Simulation of Flow and Heat Transfer in 3D Printable Triply Periodic Minimal Surface Heat Exchangers

Benjamin Wynne Reynolds

A thesis submitted for the Doctor of Philosophy in Chemical and Process Engineering

University of Canterbury

Christchurch, New Zealand

August 2020

Abstract

3D printing allows the manufacture of novel processing equipment including heat exchangers. A wide variety of different heat exchanger designs exist, and the usage of each depends on the application for which they are purposed. These designs are sensitive not only to the conditions at which they are operated, but also to shape effects caused by their internal structures. Triply periodic minimal surface (TPMS) structures were used as the internal structure of the heat exchangers studied in this work. TPMS structures tested include the Schoen gyroid, Schwarz diamond and Schwarz primitive. Shape effects cannot be easily predicted ahead of time without testing, but physical testing can be expensive when assessing the effects of many different parameters. Since 3D printing of metallic parts is a relatively difficult, developing field, numerical simulations are employed to reduce testing time and save on material costs. The lattice Boltzmann method was used as the main means of simulating the flow and heat transfer in these structures.

Designs of heat exchangers with TPMS internal structures and corresponding inlets and outlets were created and saved as stereolithography (STL) files. These files were used to store information about 3D structures and were translated into simulation domains for carrying out computational fluid dynamic (CFD) calculations. The design was not straightforward as it required a semi-automatic procedure to seal off some of the channels to ensure separation of the two heat transfer fluids. A procedure was developed to check that the design was physically feasible, that all walls were sufficiently thick, and that there was no leakage.

The lattice Boltzmann methods was set up in the Palabos environment to solve conjugate heat transfer problems in three dimensions. The problems were defined with a fluid lattice and a thermal lattice, the latter existing in the two fluid channels and the thermally conductive walls. The method was verified using previously published solutions to similar problems, in both steady and unsteady situations.

Once the method was verified, heat transfer in different representative elemental volume TPMS structures and in entire heat exchangers was simulated, while altering key parameters to determine their effects. Obtaining a stable solution within all parts of both lattices proved to be difficult. This stage aimed to increase the efficiency of the TPMS heat exchangers.

The major results of this investigation were that at the low Reynolds and Péclet numbers studied, the influence of TPMS shape effects on the heat transferred was relatively minor. The choice of geometry had a far greater influence on the pressure drop experienced by the fluid.

As such, less tortuous designs were favourable when considering a heat transfer per unit pressure drop metric. The Schwarz primitive was the best performing TPMS by this metric, though it was surpassed by a flat plate structure, designed specifically to minimise the pressure drop. When pumping is no concern and consequently the pressure drop does not matter, the gyroid was found to perform slightly better in overall heat transfer ability, as measured by the Nusselt number.

Although the current work was limited to low Reynolds numbers, future testing of TPMS designs should be carried out at higher Reynolds numbers approaching the turbulent region. At higher Reynolds number it is likely for TPMS to exhibit more significant shape effects. However, it is also likely that the lattice Boltzmann method with struggle to obtain a stable solution. In this work, only a small subset of popular TPMS geometries, in various orientations were studied. It is possible a more efficient TPMS shape than the gyroid exists, that has not been tested.

Acknowledgements

My greatest thanks go to my supervisor Ken Morison, for all his support and understanding over the course of this project. He helped guide me down the eventual path that this project took. He was always very understanding when I would want to work on just one more thing…

I also want to thank my co-supervisor Matt Watson. Although less involved in the project than Ken, Matt always had the best questions to ask, to make sure I was always thinking of something new.

Thanks to the Department of Chemical and Process Engineering (CAPE) for my three years of scholarship. It made the choice to complete a PhD a possibility, without which this never would have happened. I want to thank the CAPE technicians, particularly Graham Furniss for his help setting up my first Linux computer, and Kun Zhao for further computer help in the later years.

A big thank you to the other PhD students I had the privilege of travelling this journey with, particularly Tom Meaclem, Leatham Landon-Lane and Daniel Clarke.

The lunchtime chats with the other staff and students were often the best parts of the day…

My parents and surrounding family deserve a big thanks, both for being sounding boards for new ideas during the best of times, and amazing support during the worst. I couldn't have done this without all of you.

Table of Contents

Nomenclature

A table of nomenclature used throughout this work is included.

Symbol	Name	Units
A_{ax}	Cross-sectional area of an axial segment	m^2
A_r	Area of a radial segment	m^2
A_s	Surface area	m^2
Be	Bejan Number	
c_p	Heat capacity at constant pressure	$J\ kg^{-1}\ K^{-1}$
c_s	Lattice speed of sound	$m\ s^{-1}$
D	Diameter	m
Da	Darcy number	
dt	Lattice time scale	s
dx	Lattice cell unit length	m
Δx_w	Wall thickness	m
$\mathbf{e}_i$	Lattice directional vector	ms^{-1}
f	Fluid lattice population	
F	External body force	$kg\ m^2\ s^{-1}$
F_t	Correction factor	
$\hat{f}$	Fluid lattice population post-collision	
g	Thermal lattice population	
$\hat{g}$	Thermal lattice population post-collision	
g_{acc}	Gravitational acceleration	$m\ s^{-2}$
h	Wall heat transfer coefficient	$W\ m^{-2}\ K^{-1}$
H	Length	m
k	Thermal conductivity	$W\ m^{-1}\ K^{-1}$
K	Permeability	m^2
L	Characteristic length	m
m	Mass	kg
$\dot{m}$	Mass flow rate	$kg\ s^{-1}$
N	Lattice resolution	
NTU	Number of heat transfer units	
n	Normal vector	
n_{skip}	Lattice skip counter	
Nu	Nusselt number	
P	Pressure	Pa
Pe	Péclet number	
Pr	Prandtl number	
Q	Heat source	W
q	Heat flux	$W\ m^{-2}$

Q_m	Number of lattice directions	
Δr	Radial segment distance	m
Ra	Rayleigh number	
Re	Reynolds number	
T	Temperature	°C
t	Time	s
u	Velocity	m s^{-1}
U	Overall heat transfer coefficient	W m^{-2} s^{-1}
V	Volume of a whole	m^3
V_j	Volume of a segment	m^3
w	Lattice directional weightings	
x	Lattice position vector	
Δz	Axial segment distance	m

α	Thermal diffusivity	m^2 s^{-1}
β	Coefficient of thermal expansion	K^{-1}
δt	Dimensionless lattice time scale	
δx	Dimensionless lattice distance scale	
ε_{tol}	Error tolerance	
ε_u	Heat transfer effectiveness	
λ	Ratio of thermal conductivities	
ρ	Density	kg m^{-3}
τ	Lattice relaxation time	
T	Tortuosity	
ϕ	Porosity	
Ω	Lattice collision operator	

1 Introduction

This thesis is about the application of the lattice Boltzmann method to 3D printed heat exchangers, containing triply periodic minimal surface (TPMS) structures. Traditional heat exchangers come in a number of general designs, the most common of which are the shell and tube, and the plate heat exchangers. In most designs, two fluids of different temperatures move past each other to transfer heat without direct contact or mixing. The structures that allow for these are metals, often stainless steel or aluminium, and must be designed to transmit heat at a sufficient rate, not have excessive pressure drop and yield to physical stresses caused by thermal shock or fluid conditions.

3D printing is a developing field where structures are formed through additive manufacturing, rather than more traditional subtractive approaches. Although initially confined to polymers, 3D printing now has a range of materials that may be worked with, including metals and ceramics. Novel applications in food manufacturing (Sun *et al.*, 2015) and the construction sector (Buchanan & Gardner, 2019) are also developing. Numerous environmental studies have been carried out with relation to the energy and resource efficiencies of additive manufacturing techniques (Kellens *et al.*, 2017). Although often more resource intensive for the mass production of parts, 3D printing has advantages in the prototyping of new designs, and in areas where distributed or remote production is preferable.

Triply periodic minimal surfaces are shapes that have repeating patterns across three dimensions, and are local minima for surface area to volume ratios (Karcher & Polthier, 1996; Meeks & Pérez, 2011). This property of minimising surface area to reduce pressure drop, while still providing a tortuous path through which fluids may flow, make TPMS candidates for a variety of applications across engineering. Such applications include the development of solar cells (Crossland *et al.*, 2009), medical scaffolding structures (Ataee *et al.*, 2018), chromatography (Simon & Dimartino, 2019), and bioreactor design (Elliott *et al.*, 2017). TPMS structures are easily tunable, with alterable parameters such as the shape type, independent channel widths for each phase, the wall thickness, the rotational orientation in space and elongations or curvatures along any axis. These parameters allow for the flow fields throughout the geometry to be completely changed, in order to find parameter sets that result in favourable performance.

The LBM is a form of computational fluid dynamics (CFD), from which the first significant works began emerging in the 1950s with the introduction of the easily solvable Bhatnager, Gross, Krook (BGK) collision operator (Bhatnagar *et al.*, 1954). The LBM is based on the kinetic theory of gases, whereby the fluid is treated as a collection of particles experiencing inelastic collisions. Particles relax towards an equilibrium state over time, controlled by an equilibrium distribution function. The LBM is capable of modelling the flows passing through very small structures, and may do so with high computational efficiency due to the ease of parallelisation. Thermal effects are modelled dynamically through the simultaneous solving of a second thermal lattice. The combination of these different elements, applied to complex structures across three dimensions is unique and hadn't been reported at the start of this work.

The primary objectives of this work were to investigate the heat transfer performance and fluid flow profiles through various 3D printable structures. In addition, several novel geometries were to be tested, to check if they offered increased performance over more conventional designs. These objectives were to be achieved by creating a method by which triply periodic minimal surface (TPMS) structured heat exchangers could be formed, the writing of a program capable of simulating three dimensional conjugate heat transfer for unsteady systems, and the subsequent application of this program to the aforementioned heat exchangers. The thermal properties of the TPMS tested were to be compared, both with other TPMS, and with conventional designs.

Before the aforementioned topics could be investigated properly, first a method was required by which the desired structures could be created digitally. Chapter 3 focusses on the creation of the stereolithography (STL) files used by 3D printers. STL files consist of sets of points in 3D space, linked together into triangles. By linking many of these triangles together, any real shape may be formed, provided the triangles are sufficiently small. These same STL files required by the 3D printers could also be used as the basis for simulations. Through the appropriate voxelisation algorithms, whereby simulation cells are defined, these STL files may be reinterpreted back into the square grids used by the LBM.

Within Chapter 4, each of the numerical methods employed in this work are detailed. First a finite differencing method scheme was implemented to give a first approximation of the behaviour of the heat exchangers. The scheme was designed to be fast and simple to create, without the need to supply complex geometries to model. Instead, this version focussed on the fundamentals of the problem, making the assumption of a uniform set of flow paths. This meant

that although the finite differencing model could be created and run relatively quickly, it was incapable of assessing the characteristics of specific TPMS geometries.

In order to model the complex geometries found within the TPMS heat exchangers, a more easily extendable CFD method was required. The lattice Boltzmann method (LBM) was chosen due to the small scale of the parts being investigated, and the ability to easily parallelise simulations for greater computational speed. Previous works have made use of the LBM to simulate fluid flows through small TPMS structures, in the context of adsorption processes (Dolamore *et al.*, 2018). The present version utilised the STL files formed in Chapter 3 to give a simulation domain as close to a physical part as possible. The full-scale model was primarily used to determine the flow paths that fluids would take throughout the structures and ensure that the STL file did not contain major holes. User inputs are minimised when dealing with the STL files, in favour of automation, capable of assigning entrances and other boundaries across a range of geometries. Stability issues prevented using the full-scale method being used extensively for evaluating thermal effects. These thermal effects required the use of a representative elemental volume model, specifically unit cells of the TPMS structure. When simulating this much smaller domain, a higher resolution could be reached. The unit cells could then be used to compare the relative performance of heat exchangers consisting of them.

Once the numerical methods were chosen, the results obtained by their use needed to be verified, as is done in Chapter 5. This verification process was first carried out against known numerical benchmark cases. These results are ones obtained consistently across many studies with varying origins and methods. The second part of this chapter matches the model results against experimental data for some of the TPMS heat exchangers being studied. This too is completed in two parts: first by comparing the flow profiles formed numerically to those found by magnetic resonance imaging, then by comparing the calculated overall heat transfer coefficients.

Chapter 6 encompasses the main investigation of the performance of the TPMS heat exchangers across many geometries, orientations and materials. This process is largely carried out through the calculation of heat transfer coefficients and pressure drops for unit cells of the TPMS structure, while analysis of the entire structure is used primarily for the purpose of determining the flow profile across the many channels. The effective number of heat transfer units, and heat exchanger effectiveness are also calculated for each structure.

2 Literature Review

2.1 Additive manufacturing

2.1.1 Overview

The heat exchangers produced or modelled in the research programme will be produced by additive manufacturing. A range of materials may be used: polymers for prototyping and fluid testing, metals for heat transfer trials and ceramics for the possible application of heat transfer in reactors. Additive manufacturing is a developing technique whereby a process is used to create an entire pre-programmed 3D structure, starting from nothing. This may be achieved through the use of any of the following techniques: fused deposition modelling (FDM), inkjet printing, laminated object manufacturing, laser engineered net shaping, stereolithography printing (SLA) or selective laser sintering (SLS) (Huang *et al.*, 2012). Huang further describes each method as thus. Fused deposition modelling involves extruding a liquid thermoplastic, heated to slightly above its melting point, onto an ambient temperature substrate. Upon contact, the thermoplastic immediately sets, forming a layer that may then be subsequently built upon. Inkjet printing involves the use of an ink containing a solute material, dissolved in an organic solvent. Once released from the print head, the ink dries on the substrate due to the evaporation of the solvent. Using specially designed templates, some limited 3D structures may be formed. Laminated object manufacturing is the simple process of laminating many sheets of material, often paper, using a plastic material. Many of the sheets are stacked together, and lasers are used to cut out the shapes. Laser engineered net shaping uses lasers to create molten pools of substrate and added metal particles. The part is moved around while under the laser to create each layer. Stereolithography uses a laser to imprint a copy of the cross section of the part onto the top of a photosensitive resin. This resin is then used to coat the part, building a layer. Selective laser sintering uses a laser to sinter together fine particles of the build material. Each layer of the object is melted into the particle bed one at a time, building up the object.

Although additive manufacturing began emerging in the 1980s (Kruth, 1991), advancement in the field has only recently undergone a period of extreme growth (Huang *et al.*, 2012). This growth has been partly due to the lowering costs of purchasing 3D printers, and partly the rise of the internet, allowing for enthusiast communities to emerge and generate creative ideas. The freedom to design objects from home, in programs such as SOLIDWORKS, has garnered the attention of the public, and spurred a large volume of research.

The process promises major advantages over the currently dominant subtractive manufacturing. Subtractive manufacturing uses drills, lathes and computer numerical control machines to remove material from a solid sheet or block, in order to reach a desired shape. Additive manufacturing instead builds the shape layer by layer, starting from nothing. Although waxes are often used in temporary support structures, these are much cheaper than the bulk material in use. Across a wide range of industries, there is a potential for 3D printing to give a faster and more accurate build, at a potentially lower cost (Buchanan & Gardner, 2019). While the material selection was not within the scope of this thesis, it was necessary that simulations were based on physically realisable structures. The sections following describe a number of feasible materials, some of which are investigated more fully in further chapters. A few of the basic thermal properties for some of the relevant materials are given in Table 1.1.

Table 1.1. Thermal properties of materials relevant to 3D printing.

Thermal Properties	ABS	PLA	Stainless Steel	Aluminium	Water
Thermal Conductivity (W/m.K)	0.163	0.13	14.6	226	0.603
Density (kg/m^3)	1070	1240	7920	2700	1000
Specific Heat Capacity (J/kg.K)	1990	1800	502	921	4180
Thermal Diffusivity (mm^2/s)	0.0766	0.0582	3.68	91.0	0.144

2.1.2 Polymers

Likely the area of additive manufacturing with the most development is the printing of polymers. Unfortunately, most polymers exhibit poor thermal properties, making them unsuitable for applications where heat transfer is important. This downside makes polymers more suited to the roles of prototyping parts for later creation with other materials, or for applications closer to room temperature, where the tensile strength of a metal is not required.

Foremost amongst the polymers used in 3D printing are polylactic acid (PLA) and acrylonitrile butadiene styrene (ABS). PLA prints are typically the easiest to print, and give the least warping during the printing process. This makes them ideal for simply prototyping, or consumer use. PLA is however restricted to applications at approximately room temperature, due to its relatively low glassing temperature. ABS is stable across a much wider range of temperatures, potentially reaching as high as 100 °C before the glassing temperature is met (Crews *et al.*, 2016), though produces lower quality results. SLA prints make use of photo-polymer resins, which are cross-linked into solids using UV lasers, allowing for parts to be printed with great precision (Ligon *et al.*, 2017; Wu *et al.*, 2020). The photo-polymers used are

complex proprietary blends made from a range of compounds, which may be customised to give better performances in mechanical strength, aesthetics, or thermal resistance (Steinmann, 2006).

One popular idea being trialled across a number of studies, is the premise of combining the easy to print ABS, with a catalytic coating capable of carrying out reactions. ABS may be readily used as a support for catalysts due to its porous nature. Coating an ABS skeleton in copper benzene tricarboxylic acid (Cu-BTC) and drying, results in a catalyst capable of effectively removing methylene blue (Wang *et al.*, 2014). The catalyst is durable and precisely reproducible, due to the 3D printed construction. A common issue with such designs is the challenge in the fouling and regeneration of such a catalyst. While the issue may not be significant for basic geometries, some areas of complex structures more complex designs will not facilitate even transport across the entire geometry, reducing the effective, working surface area available over time.

3D printing has been used to construct the packing of chromatography columns (Fee *et al.*, 2014). Using ABS plastic with paraffin wax supports, straight and fish-bone shaped channels could be created to micron precision. Knowing the exact position, size and volume of the pores in the packing gives a very high level of control, allowing for better optimisation of the performance of the column. A similar approach could be taken to other separation process that normally rely on random packing materials, such as gas scrubbing, distillation, demisting, adsorption and catalysis. While these applications do not normally involve heat transfer, the mass transfer within them is directly analogous to heat transfer, so the simulation techniques developed here are likely to be applicable.

For polymer printing, the minimum stable layer size possible for most current 3D printers is approximately 50 μm (Wu *et al.*, 2020), though this varies with the particular printing technique employed, the materials, and the actual printer used. Since singular layers are not mechanically stable, features must typically consist of at least two fused layers, though usually more in order to prevent warping. Although physical features must therefore usually be at least 100 μm in size, the accuracy to which the layers may be placed is an order of magnitude lower. Fee *et al.* (2014) for example made use of an ABS printer to carry out fused deposition modelling on octahedral beads of 115 μm apogem, to a print resolution of 28 μm.

2.1.3 Metals

For heat exchangers, a metallic structure would normally be used. Metallic alloys must be carefully selected, as these must be brought to very high temperatures to be melted during additive manufacturing. At these temperatures, partial vaporisation of some components of the molten alloy can occur. This vaporisation can be deleterious to the print, as the composition of the alloy is changed. Changes to the composition of the alloy will almost invariably lead to inferior characteristics of the part, including degradation to the hardness, corrosion resistance and tensile strength (Mukherjee *et al.*, 2016).

Despite being less developed than the plastics field, the number of metals that can be 3D printed is growing. The list includes various alloys containing stainless steel (Yang *et al.*, 2017), copper and aluminium (Tubio *et al.*, 2016); titanium (Szost *et al.*, 2016); silver (Cooperstein *et al.*, 2015; Lopes *et al.*, 2014); gold (Bieri *et al.*, 2003); bismuth indium, tin and zinc (Wang & Liu, 2014). Printed metallic structures have the advantages of greater structural strength than plastics, and may have a higher tolerance to heat, depending on the alloy.

With the proper choice of metallic media, entire conductive structures or narrow conductive pathways can be printed. Conductive inks are usually made up of a less conductive bulk material, containing dispersed conductive nanoparticles made up of metals such as silver (Cooperstein *et al.*, 2015) or gold (Bieri *et al.*, 2003). An alternative method under development is the inclusion of carbon black in 3D printer inks, which at volume concentrations above 25% becomes electrically conductive (Leigh *et al.*, 2012). Carbon black has the advantage of costing very little, since it is an amorphous form of carbon created as a by-product in several industries. When these conductive inks set, conductive pathways, or even entire conductive structures may be formed.

The opposite of the inclusion of conductive pathways, is the inclusion of empty channels throughout the structure. Through such channels, coolant or heating fluids may be delivered to all positions within the walls, supplying a heating duty from within. A small number of channel shapes have been trialled in the past, for the cooling of a mirror (Mici *et al.*, 2015).

2.1.4 Ceramics

The potential exists to combine heat transfer and reactions with ceramic materials. Ceramics are typically more difficult to additively manufacture than polymers or metals, largely due to their inherent brittleness and difficult to control porosity (Eckel *et al.*, 2016). The techniques

that are still available for use include: jetting a binder agent into a particle bed, to bind together ceramic particles; using a laser to selectively fuse ceramic particles; or the use of a polymer-derived ceramic. As described by Eckel *et al.* (2016), these polymer-derived ceramics may be constructed by adding organic groups such as thiols or epoxies onto a ceramic backbone. The organic functional groups make the resin workable, such that the 3D printer may construct the desired geometry. The shaped resin is subsequently pyrolysed to remove the organic groups. Provided that sufficient ceramic material was present in the resin, the remaining ceramic will keep its shape, yielding a solid 3D printed ceramic structure.

Many metal oxides are used as heterogeneous catalysts. Being oxides, they are relatively resistant to temperature and pressure effects, while often being easily accessible materials. Copper in particular, due to the diverse range of reactions it may be used in, and low cost, is an especially attractive catalyst. 3D printed structures have been made from inks comprised of catalytic materials. Tubio *et al.* (2016) describe a procedure to create a Cu/Al_2O_3 catalyst, which may subsequently be loaded into an inkjet type 3D printer, and formed into a wood stack shape. The wood stack is afterwards sintered in a furnace, before it may be used as a functioning catalyst. The copper component is used as the catalytically active species, embedded within an Al_2O_3 support structure. By 3D printing the catalyst onto the support structure, exact control is possible over the total and localised catalytic loadings. Catalyst research in particular would benefit from an added level of control over the catalyst distribution within a structure being studied.

2.2 TPMS

The current broader programme of work with which this project sits is focused on triply periodic minimal surface structures. Minimal surfaces are those which display a local minimum of surface area per unit volume (Jung *et al.*, 2007). They were first investigated in the 1800's by physicists studying the shapes naturally taken by soap bubbles (Karcher & Polthier, 1996). Bubbles tend to minimise their surface areas in order to minimise free energy. Triply periodic minimal surfaces are minimal surfaces periodically repeating across three dimensional space, with no straight lines or self-intersections (Osserman, 1986; Schoen, 1970). Many TPMS may also be classified as infinitely periodic minimal surfaces. Physically, the two categories are the same, though mathematically some surfaces will continue repeating regardless of the number of dimensions upon which the pattern exists. The single gyroid has an infinitely repeating morphology, where the two phases are separated into two distinct channels, each taking up half

of the total volume of the structure (Schoen, 1970). As denoted by Miller indices, these channels run in the (100) and (111) directions, and exist across multiples of the unit cells shown in Figure 2.1. Two forms exist for the gyroid, and most other TPMS geometries: a skeletal form, and a sheet form. In the skeletal form, a single fluid phase and another single solid phase exist. The two are in contact with each other, with no other separation. Skeletal forms have been investigated in adsorption related processes (Dolamore, 2017), where previously designed packings have had choices between structured or random packings.

The sheet form allows for two separate fluid channels to exist, separated by a solid, relatively lower in volume. The two channel systems do not intersect, and as such may potentially be used in applications where fluids are not allowed to mix. One example is the heat exchanger, where the channels are usually highly ordered, with the purpose of creating highly curated flow profiles.

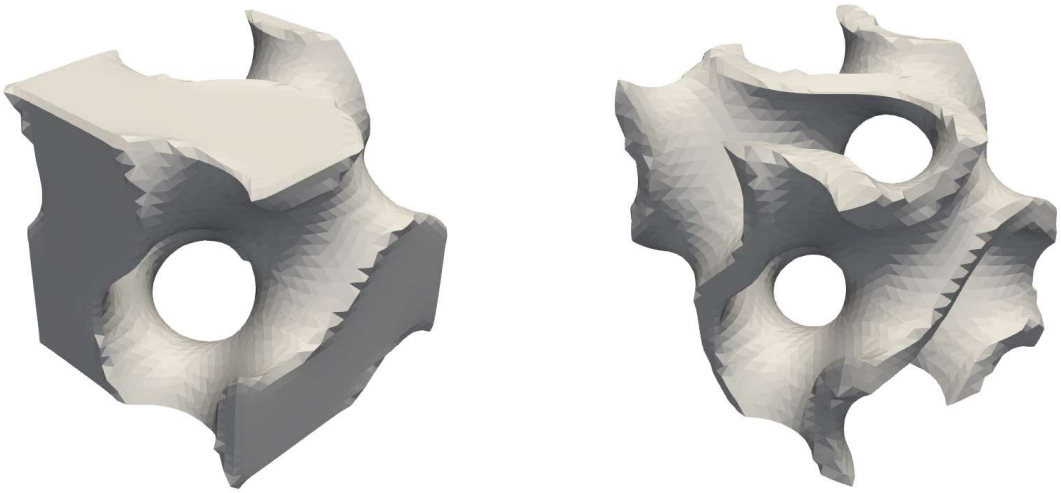


Figure 2.1. Representation of the structure of A) a skeletal gyroid and B) a sheet gyroid.

The shape is a form of intermaterial dividing surface, which are those geometries that divide a volume in three dimensional space, into two or more distinct domains. The generalised form for all such surfaces are given by (Scherer, 2013):

$$F(x,y,z) = \sum_{hkl} |F(h,k,l)| \cos\left(\frac{2\pi}{L}(hx + ky + lz - \alpha_{hkl})\right) \tag{2.1}$$

where x, y and z are cartesian coordinates, h, k and l are the periodicity in each direction, and α is an offset. The function $F(h,k,l)$ is a tuneable expression, giving the shape of the particular structure being formed. The quantity L is the length of the unit cell. Using the lowest allowable values of the parameters h, k, l and α, the level surface for the single gyroid may be formulated:

$$f(x,y,z) = \sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right) + \sin\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) + \sin\left(\frac{2\pi z}{L}\right)\cos\left(\frac{2\pi x}{L}\right) = t \tag{2.2}$$

By varying the value of t, the properties of the gyroid structure can be altered. When $t = 0$, the result is known as the minimal gyroid. The two phases each take up 50% of the total volume and the mean curvature of the shape is zero. When t takes values between 0 and 1.413, the volume of one of the channels grows with respect to the other. For values of t between 1.413 and 1.5, parts of each structure are no longer connected, making the structure lose much of its functionality. For values of t above 1.5, no set of coordinates may satisfy the equation, so the shape does not exist. These values are unique to the single gyroid. Other TPMS have different breakpoints for which the parameter t may make the geometry infeasible. The sheet gyroid makes use of the same equation, employing two separate spatially parallel surfaces, each with their own t value offsets from a central fictitious surface location. These offsets therefore become one of the parameters used to tune the wall thickness of any generated geometries.

The gyroid, amongst a total of 17 infinitely periodic minimal surfaces, was first discovered by Schoen (1970), while working for NASA. The morphology has circular tunnels running in the (100) and (111) directions, as defined with Miller indices, with no straight lines or plane lines of curvature. Schoen remarked that the gyroid surface G, appeared to be a hybrid of the Schwarz P and D surfaces, and was unique in that it was the only surface found to create oppositely congruent labyrinths.

These same properties apply to a wide variety of TPMS structures. Other shapes of interest are those originally discovered by Schwarz (as cited by Schoen, 1970), to exist without self-intersections. These include the Schwarz primitive, diamond, hexagonal, complementary primitive and crossed layers of parallels surfaces. This lack of self-intersection was not able to be mathematically proved until the work of Große-Brauckmann and Wohlgemuth (1996). Figure 2.2 shows an example of the skeletal and sheet forms of the Schwarz diamond structure. The skeletal form consists of two phases only, with each phase completely interconnected within itself. Sheet gyroids contain three labyrinths, one made up of matrix material, and two separate, non-intersecting channels. The diamond configuration is named after the shape that each of the channels makes within the structure (Schoen, 1970). The tetrahedral pathways are similar in shape to the chemical bonds that form the links within a diamond network.

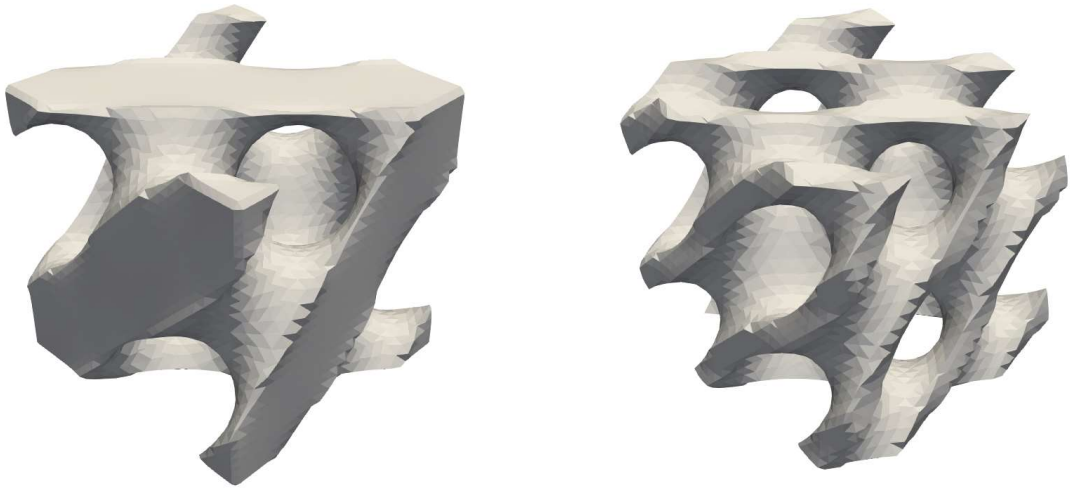


Figure 2.2. Representation of the structure of A) a skeletal diamond and B) a sheet diamond.

All TPMS may be formulated in one of three ways: Enneper-Weierstrass integration, as was used by Schwarz and Schoen; optimisation of the Landau-Ginzburg functional; or Fourier series approximation of periodic nodal surfaces (Jung *et al.*, 2007). Fourier series approximation has become the most common method due to its simplicity and relative accuracy to the true minimal surfaces, even with relatively few Fourier terms (Gandy *et al.*, 2001).

The equation for a Schwarz diamond, as formulated in its level-set form by von Schnering and Nesper (1991) is more complicated than that of the gyroid. In addition, more than one form exists for the diamond equation. The version used throughout this work is given in Equation 2.3:

$$\begin{aligned} f(x,y,z) = &\sin\left(\frac{2\pi x}{L}\right)\sin\left(\frac{2\pi y}{L}\right)\sin\left(\frac{2\pi z}{L}\right) + \sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) \\ &+ \cos\left(\frac{2\pi x}{L}\right)\sin\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) \\ &+ \cos\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right)\sin\left(\frac{2\pi z}{L}\right) = t \end{aligned} \tag{2.3}$$

Many variations to this surface may be found by swapping the sin and cos terms or otherwise altering the phase of each term. Generally these will produce rotated, but otherwise equivalent surfaces. TPMS have classically been segregated into genera in which any member may be changed to another through simply deforming the structure. Such surfaces are topologically similar. These surfaces and the gyroid each belong to the same genus (Schoen, 1970), making them relatively similar to each other mathematically. The Schwarz primitive differs from the

generic gyroid and diamond in that the exposed surface area for each channel is not guaranteed to be equal, for any arbitrary slice. The Schwarz primitive is shown in Figure 2.3:

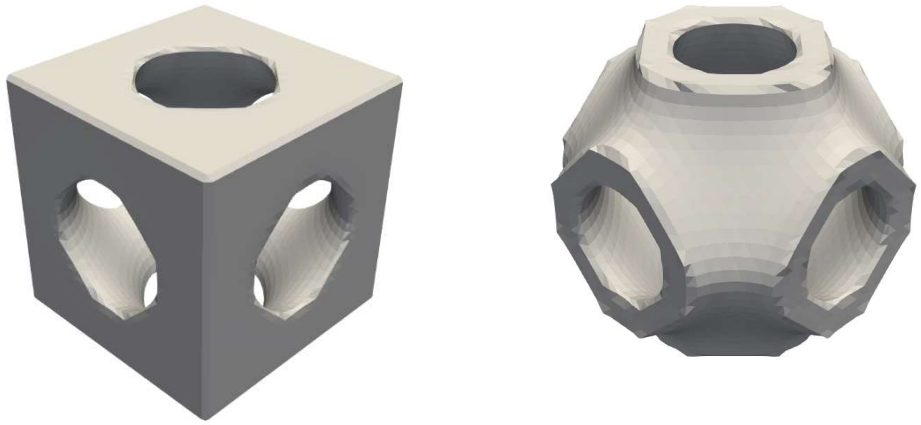


Figure 2.3. Representation of the structure of A) a skeletal primitive and B) a sheet primitive.

The equation for a Schwarz primitive surface is perhaps the most simple of the TPMS. It is given by Equation 2.4 (von Schnering & Nesper, 1991):

$$f(x,y,z) = \cos\left(\frac{2\pi x}{L}\right) + \cos\left(\frac{2\pi y}{L}\right) + \cos\left(\frac{2\pi z}{L}\right) = t \tag{2.4}$$

Composite TPMS geometries may be formed in one of two ways. The first method is to manually join together shapes in alternating, discrete layers. Although many applications exist in which one geometry is strictly better than others, the area of transition between geometries is capable of creating flow distortions. These regions may be used to break up laminar flow, ensure better mixing, or provide structural integrity. Another form of composite structure may be formed through carrying out basic arithmetic operations on combinations of TPMS equations (Dolamore, 2017). One example of such an operation is the simple addition of the equations for each of the Schwarz primitive and the Schoen gyroid, as demonstrated in Equation 2.5.

$$\begin{aligned} f(x,y,z) = {} & \sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right) + \sin\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) \\ & + \sin\left(\frac{2\pi z}{L}\right)\cos\left(\frac{2\pi x}{L}\right) + \cos\left(\frac{2\pi x}{L}\right) + \cos\left(\frac{2\pi y}{L}\right) \\ & + \cos\left(\frac{2\pi z}{L}\right) = t \end{aligned} \tag{2.5}$$

Dolamore found that some combinations of TPMS provided levels of chromatographic efficiency similar to that of basic TPMS. Figure 2.4 gives examples of each of the two forms of composite TPMS geometry.

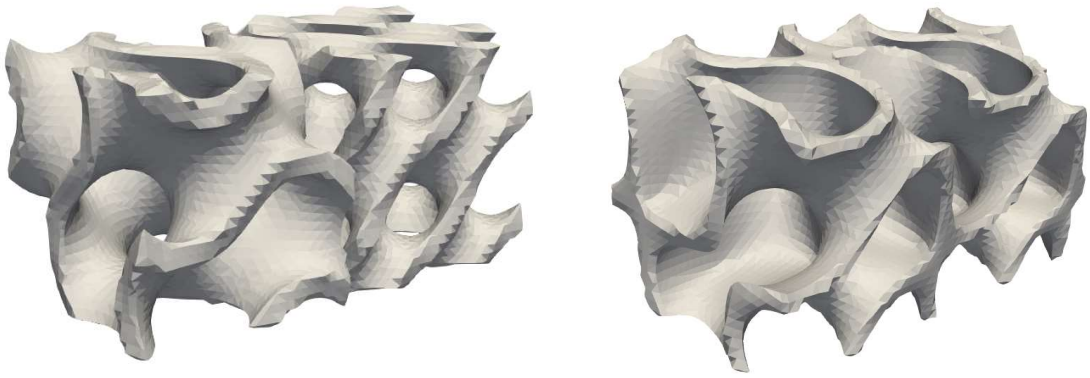


Figure 2.4. Composite TPMS structures. A) Discrete layers and B) Arithmetic combination.

When discrete layers of TPMS are employed, applications must be restricted to those using a single fluid only. Complete partitioning of the fluids cannot be guaranteed at the cell boundaries, and attempts to do so through the placement of channel blockers would significantly inhibit fluid movements. Layering of different TPMS may be advantageous in applications where the spatial performance of a unit operation is to be optimised. The flow profile through a structure may be made more uniform through the deliberate placement of TPMS packing materials that are better performing, but with higher pressure drops. The range of alternative structures that may be formed through the use of arithmetic combinations of the TPMS equations is almost limitless. TPMS are the local minima of the surface area to volume ratio, but arbitrarily convoluted structures are not necessarily so. Jung *et al.* (2007) were able to modify the equations for various TPMS, and then optimise towards new surfaces, however these new surfaces were not achieved instantly.

Although 3D printing is the primary focus of this work, TPMS structures may be formed by other methods, or through natural means. In the correct molar ratios, temperatures and pressures, oil–water systems may self-assemble into gyroid morphologies. This occurs due to the gyroid configuration becoming the thermodynamically favoured geometry (Scherer, 2013). It follows that further changes to these physical properties can prompt a shift back out of the gyroid, into another structure such as the Schwarz diamond (Hyde *et al.*, 1984). Phase diagrams have been constructed for some two component systems to demonstrate the effects of composition and temperature on the preferentially formed phase and the resulting morphology formed.

Carbon is another material that may be formed into a gyroid network. Werner *et al.* (2014) were able to form both single and double gyroid structures, made up of mesoporous carbon, within which the porosity and pore sizes were easily tuneable. The material requires extended curing under inert atmospheres, there are no foreseeable applications for this material in the field of 3D printing. Applications may exist however, when used as a high surface area electrode, or catalyst substrate.

Diblock (Ma *et al.*, 2016; Matsen & Schick, 1994; Vukovic *et al.*, 2012) and triblock (Avgeropoulos *et al.*, 1997) copolymers may be used to form gyroid networks. These cases, whereby polymers naturally form into TPMS structures, demonstrate the physical meaning of Equation 2.1, due to the structures representing local energy minima. Diblock copolymers are composites that are made up of two entirely separate polymers that have been joined together in the centre. Similarly, triblock copolymers are made up of two units of one polymer, and one unit of another, in an alternating chain. If the two types of polymer are chosen such that they are not miscible with each other, the parts will tend to separate into different phases. Polymers with large molecular weights have enough degrees of freedom of movement to do this. Since the polymeric blocks are still joined, complete separation of these phases is not possible, leading to the creation of a minimal surface. In some cases, this minimal surface will be a gyroid, if it is the thermodynamic minimum of energy. Also possible are the diamond, double diamond, cubic, hexagonal, lamellar, catenoid lamellar and many more phases (Matsen & Schick, 1994). The gyroids formed by this block copolymer technique may form structures on the nano-scale, useful for catalysis, amongst other separation processes (Cheng *et al.*, 2015).

Analogous to the use of diblock copolymers, Phillips *et al.* (2010) found that a polymer joined onto an immiscible metal nanoparticle may also yield a double gyroid. They reportedly used CdTe nanoparticles, although Au and SiO_2 particles were used by others researching a similar topic. The metal-polymer chains form into gyroids to minimise their respective energy states, keeping the nanoparticles and polymers as far apart as possible. In the correct compositions, the gyroids formed were made up of a polymer support, and a metal surface phase. It is possible, that with the replacement of the polymer with a metal of higher thermal conductivity, a structure suitable for heat exchange could be formed. Such a structure would gain the desirable thermal properties from the metals, while maintaining the reactive capabilities from the nanoparticles.

Gyroids have far reaching applications, from medical applications such as neural electrodes or drug delivery systems (Cho *et al.*, 2015), to catalyst supports, nanowires or high surface area

electrodes. One method used by Urade *et al.* (2007) involved electrodepositing platinum onto the surface of a calcined diblock polymer surfactant. The gyroidal platinum was then useful for nano electronics, use as an electrode, or for catalysis. Ma *et al.* (2016) carried out a somewhat similar process, instead depositing layers of ZnO by atomic layer deposition. The thickness of these layers could be controlled tightly, leading to similar applications in electronics or catalysis. Cho *et al.* (2015) formed a biocompatible, conductive gel into the gyroid morphology, before further polymerising and testing.

Another potential use for TPMS geometries is as a tool for carrying out controlled chemical reactions. In the reaction:

$$A + 2B \rightarrow C \tag{2.6}$$

the two components A and B are required to mix thoroughly, and in the correct ratio to form the product C. Some reactions where this is the case are simply carried out using the correct molar amount of A, and an excess of B (Davis & Davis, 2003). However, this leaves the product C contaminated with a large amount of the excess B, which could be uneconomical, dangerous or detrimental to the product. One solution would be the creation of a new split gyroid, whereby similar to sheet gyroids, two main non-connecting flow channels would permeate the structure. An additional channel would be formed in the middle of the ordinary wall space, splitting it in two, and leaving a third channel between the original two. By choosing the wall materials carefully, the channel containing A could be made permeable to A, while the B channel would be permeable to B, diffusing at twice the molar rate than the original channel did for species A. Once in the new reaction channel, the products may be removed through a pressure differential, gravity, or some other force. Simulation of such a system dominated by mass transfer would be analogous to an equivalent heat transfer simulation. Alternatively, the central channel may remain impermeable to all compounds in use, and instead be used strictly for temperature control.

Many successes have been had regarding incorporating TPMS into the structure of chromatography columns. Research into structured packings is ongoing, with significant progress being made recently into the 3D printing of column packings. Fee *et al.* (2014) were able to create an exactly reproducible packing structure and characterise it with residence time testing. Due to the limitations of 3D printing technology at the time, the channels created in their packing could only be straight or fishbone shaped, with relatively large diameters, at around 200 μm.

With 3D printing in mind, although the scale and operating conditions differ, direct comparisons can be made between the potentials for chromatography columns and heat exchangers. Each hold their fluid contents for some amount of time, passing it over internal surfaces before releasing it, ideally without incurring an excessively large pressure drop in the process. Where chromatography aims to reach a minimum plate height, leading to maximum separation efficiency for a corresponding pressure drop, heat exchangers aim for a set amount of heat transferred between channels, or a corresponding temperature change, again at the expense of pressure. For this reason, work into chromatography structures is highly relevant to heat exchangers, where the concept of 3D printing is just beginning to catch on.

2.3 Heat Exchangers

2.3.1 Basic designs

A heat exchanger is a unit used to transfer heat between one or more fluids of different temperatures without direct contact between them. This operation is carried out on a fluid with the goal of heating, cooling, or recovering waste heat. The two fluids may be of different composition, phase, flow rate or other thermal properties such as thermal conductivity or heat capacity. The heat exchanger is a well-developed industrial unit, and as such, heat exchangers have taken on a wide variety of designs, depending on the application. Factors affecting the choice of design include material cost, pressure drop, ease of cleaning, structural integrity and heat transfer coefficient (Kuppan, 2013). Kuppan further details many of the common heat exchanger designs.

Typically, heat exchangers are constructed from metals with high thermal conductivities such as steel, aluminium or sometimes copper. These metallic walls are made as thin as possible to aid in the transfer of energy between the fluids. Doing so brings the fluids in as close thermal proximity as possible to transfer the most heat without the prospect of mixing. As described by Kuppan (2013), the factors influencing the wall thickness include pressure, corrosion resistance, mechanical wear, axial strength, use of standardised dimensions and cost. Typically units will be built using standardised parts according to the Tubular Exchanger Manufacturers Association (Tubular Exchanger Manufacturers, 1999) guidelines. Tubes vary from 6-50 mm in outer diameter, with wall thicknesses generally on the order of one tenth of the total outer diameter of the tube.

Perhaps the most widely used form of heat exchanger is the shell and tube design (Lee, 2010). Two fluids of different temperatures enter. One fluid is split amongst the many small pipes of a tube bundle, while the other fluid fills the shell surrounding the tubes. Depending on the shell type, the tube bundle may either pass through the shell once only and emerge from the opposite side, or if greater heat transfer is required, be turned back around and make a second pass (Poddar & Polley, 2000). In this case the tube fluid enters and exits on the same side. The same number of tubes exist, but the flow is split amongst half as many passing through each direction. This increases the velocity of the flow, increasing the heat transfer to the exchanger walls. To further optimise the design, many other features, such as wire inserts into the tubes, helical baffle plates, and both internal and external fins may be added (Yang *et al.*, 2020). Figure 2.5 demonstrates a simple counter-flow example.

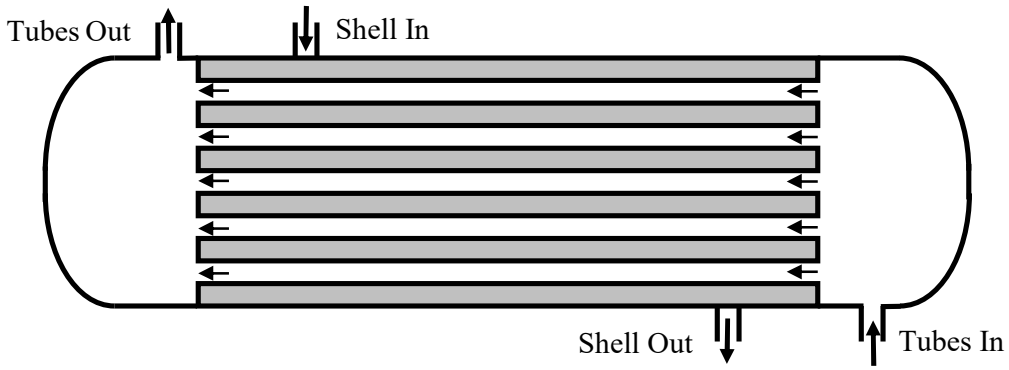


Figure 2.5. Cross-sectional view of a single pass shell and tube heat exchanger.

Another very widely used design is the plate heat exchanger. As is described in greater detail by Manglik *et al.* (2007), corrugated metallic plates are held together by a frame that allows each fluid to access alternating plates. As such, the sides of each plate separate the hot and cold channels traversed by the fluids. Rubber gaskets are placed on alternating plates to block ingress by the incorrect fluid. Both single and multi-pass variants of the plate heat exchanger exist. Figure 2.6 shows a simple example of a plate heat exchanger. The hot and cold streams are split amongst each of the plates. Fluids need only pass over a single plate before accessing the outlet.

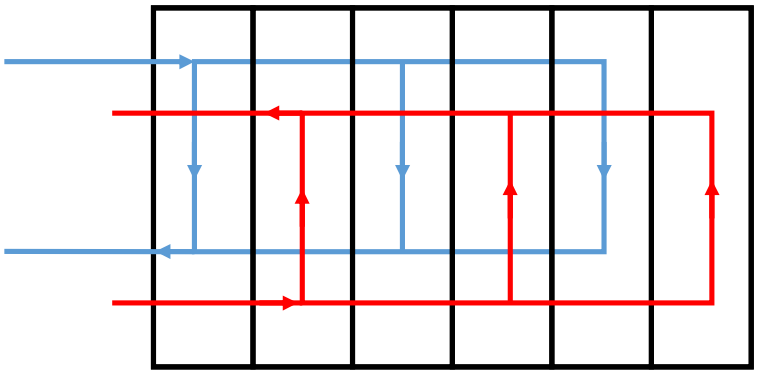


Figure 2.6. Diagram of flows in a plate heat exchanger.

Each of the plates is pressed with corrugations in the shape of chevrons, of which the angle may be altered to influence the flow profile over the plate (Khan *et al.*, 2010). Construction of the plates is simple, as the sheets of metal merely need to be pressed into shape and have gaskets put in place. Plates may be added or removed to holding frame to change the available surface area. The surface area of a plate heat exchanger may be far higher than that of a shell and tube design of equivalent size. This can be impacted by the number of plates, the spacing between the plates, plate thickness and the angle and depth of the corrugations.

One variant of the plate heat exchanger is the plate and frame. This version consists of many plates held within a frame more resembling the exterior of a shell and tube heat exchanger. One fluid stream is piped through the plates as in the plate design, while the second fluid is free to move through the shell, as with the shell and tube design. This construction is significant as it represents a version where the fluids move through incongruous pathways through a labyrinth. The fluid movement is somewhat analogous to that passing through a form of packing consisting of discrete and repeating unit cells. As with porous media, the performance of the layout may be improved by altering the either the physical characteristics of the individual unit cells, plates, contained within, or by making changes to the greater macrostructure (Najafi & Najafi, 2010). Such optimisations affect both the overall heat transfer coefficient, and the pressure drop.

Another heat exchanger variant with similarities to those formed from TPMS is the plate and fin heat exchanger. In this case, fluids pass through uniform, periodic fin structures, separated from each other by a dividing plate. The fin packing is designed to spread heat as evenly as possible throughout the structure, and expose as much surface area as possible to each of the fluids. At that point, heat may then transfer between the adjacent layers of the device exposed to other fluids. Fins are usually designed in simple, easy to manufacture geometries such as

squares and triangles, though more complex morphologies are possible. Optimisation of the fins, as with other designs, depends on the surface area, pressure drop, and heat transfer rate between the fluids (Najafi *et al.*, 2011). A simplified form of this design is the heat sink, as may be found on an electrical component. Heat sinks on electronics typically utilise only a single fluid, usually air. Convective currents are formed, forcing heated air up away from the device, allowing access for fresh coolant to the exchanger surface. Often additional fans will be positioned to blow air by advection past the heat sink, reducing the size of the heat transfer boundary layer, and speeding transfer.

Although the internal structure is perhaps the largest contribution to the heat transfer ability of a given heat exchanger, another factor of large importance is the flow configuration within it. When the fluids of each channel travel in the same direction, parallel-flow is achieved. Fluids moving in opposite directions are in counter-flow operation, while fluids moving perpendicular to each other are in cross-flow. Although localised regions of each flow type may be present within the same exchanger unit, one type is usually considered to dominate the others. A correction factor may be applied to account for such regions (Kuppan, 2013).

In parallel-flow, the fluids in each of the two channels usually enter the heat exchanger near each other, creating a large temperature differential around the inlet region (Kuppan, 2013). While this allows for relatively fast heat transfer initially, the streams will tend to approach a midpoint temperature across the length of the unit, and the rate of heat transfer will diminish. The wall temperatures will tend stay an approximately equal value throughout the length (Lee, 2010). As such, parallel-flow tends to be more suitable for applications where one fluid does not significantly change in temperature, due to heat capacity, density or flow rate differences. Alternatively, when the outlet temperatures must be controlled more tightly, parallel-flow may be used to force both streams to similar temperatures, or prevent one stream from passing a critical threshold, through the design limitations of parallel-flow.

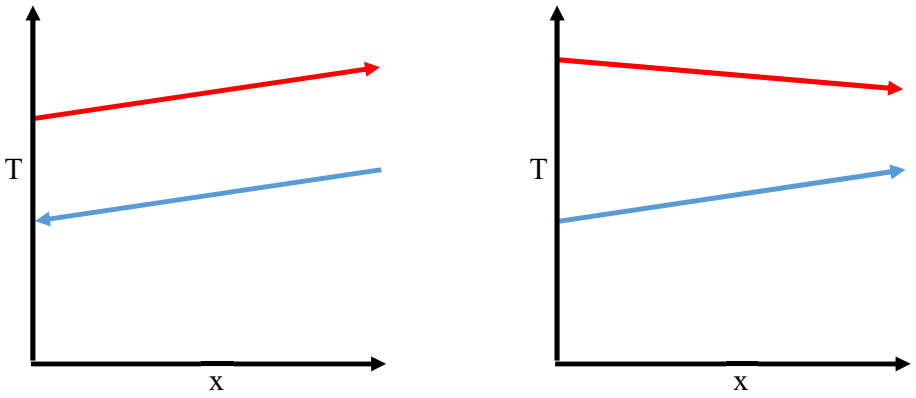


Figure 2.7. Temperature profiles for two fluids through a A) counter-flow and B) parallel-flow heat exchanger.

In counter-flow operation, the two fluids enter on opposite sides, and move in opposite directions to each other (Lee, 2010). Each of the fluid inlets is exposed to an outlet that has already undergone the full heat transfer of the exchanger. As such, the temperature differential between the streams tends to be a more consistent value across the entire length, and will average to a greater value than for parallel-flow flow. Figure 2.7 provides a diagram of the parallel-flow and counter-flow operation modes. The lower temperature gradients across the channels will also result in lowered thermal shock for streams with significant temperature differences. For these reasons, counter-flow operation is often preferred.

The other major mode of operation is cross-flow. Although not employed as much industrially as the other configurations, cross-flow is widely used in consumer products, in applications where the coolant fluid is available at no cost. One example of this is the car radiator (Bury *et al.*, 2010). Tubes filled with glycol-based fluids circulate between the engine, and the radiator area, where they are exposed to air entering from outside the vehicle. Although air has a low thermal density, very high velocities may be reached from the movement of the vehicle, providing the cooling required to prevent overheating (Kuppan, 2013).

Often some localised elements of cross-flow are present even in otherwise parallel-flow or counter-flow flow operations. When standardised shell types are used, it will be known in advance whether or not the flow is likely to approach that of pure counter-flow (Poddar & Polley, 2000). One feature that will reduce the proportion of counter-flow is the baffles within the shell of a shell and tube heat exchanger. Upon being blocked by the baffles, the fluids will move perpendicular to the tube fluids for a time, until a path around the baffle may be reached. Since multiple baffles may be present within the same heat exchanger, much of the length of an otherwise counter-flow vessel may actually be run in cross-flow operation. Such an example

is shown in Figure 2.8, where a significantly tortuous path is forced, indicating an approximately equal cross-flow to counter-flow length.

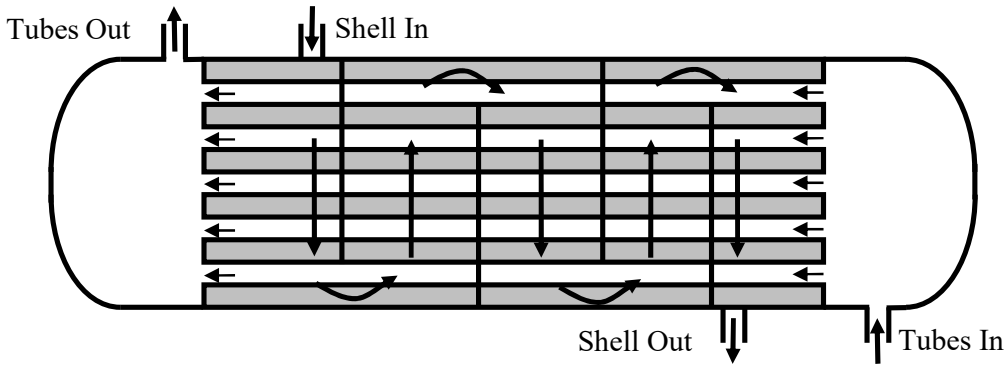


Figure 2.8. Fluid flows past the baffles of a shell and tube heat exchanger.

An alternative design to that depicted in Figure 2.8 is a unit consisting of two congruent, and otherwise equal TPMS labyrinths, instead of the highly restricted tubes and relatively open shell side present in the shell and tube design. Such a unit would have far more tortuous channels compared with a single-pass shell and tube heat exchanger, and potentially better opportunities for heat transfer from an additional degree of freedom of movement for the previously tube-side fluid. Similar to the aforementioned example, elements of cross-flow will be present in some locations throughout the structure. One potential application for a TPMS design is in the heat sink, as described earlier, whereby the cooling fins would be replaced by a TPMS structure. Doing so may increase the surface area available and improve mass transfer of cool air to the exchanger surface. In the field of microfluidics, where the scale of features is on the order of micrometres, Wang *et al.* (2018) reviewed the cooling options open to the developing field of 3D integrated circuit boards. Many still rely on designs resembling more traditional heat sinks, to which improvements in the form of TPMS could be made.

Although most heat exchangers operate at relatively high Reynolds numbers in the turbulent region of $Re > 3000$, examples of low Reynolds flows exist as well. Saatdjian *et al.* (2011) studied an annular rotating heat pipe under Stokes conditions, $Re < 1$. They aimed to increase the rate of heat transfer between phases through the use of chaotic advection caused by rotating each of their two cylinders opposite to each other. The induced chaotic mixing emulated turbulence despite the viscous flow, resulting in uniform temperature profiles and heightened rates of heat exchange. At a higher, but still laminar Reynolds number, Erbay *et al.* (2017) examined a traditional fin and tube heat exchanger operated between Reynolds numbers of 200-600 in the context of an air ventilation system. Huisseune *et al.* (2015) also investigated

ventilation systems, though in the form of a metal foam heat exchanger, and at conditions of $Re < 2000$. They replaced the fins from earlier designs with a foam instead, which the heat exchanger tubes would pass through. They compared their results to one of their earlier works (De Schampheleire *et al.*, 2013), also in the field of air ventilation, for which low Reynolds numbers are common. Their metal foams performed far better than their earlier traditional finned designs, due to the ability of the foams to form into microchannels of much higher surface area.

Prior to the beginning of the present work, only a single prior mention of a TPMS heat exchanger was identified (Yoo, 2015), in which a small number of polymer based heat exchanger designs were printed, but left untested. Before the conclusion of the present work, one additional work in the field of TPMS heat exchange has since been published. Recently, Luo *et al.* (2020) examined the thermal conductivity and permeability of sheet gyroids over a range of porosities using simulations. This work was limited to a single orientation of the gyroid TPMS, making use of representative elementary volumes. As such, no macroscopic heat exchanger designs were identified in this work.

2.3.2 Performance metrics

A number of criteria are required to analyse the performance of heat exchangers, across a range of temperatures, flow conditions and structural designs. These metrics must be valid across a range of temperatures, flow rates and designs. The logarithmic mean temperature difference (LMTD) is one of the measures used to quantify the average temperatures across the entrances of a two-fluid heat exchanger. The LMTD is defined as (Kuppan, 2013):

$$\Delta T_{LMTD} = \frac{\Delta T_A - \Delta T_B}{ln\Delta T_A - ln\Delta T_B} \tag{2.7}$$

where A is one end of the heat exchanger, and B is the opposite end. For the example counter-flow heat exchanger in Figure 2.7, the hot primary channel enters left and exits right. The cold secondary channel enters right and exits left. Although the exits are perpendicular at the ends, for the majority of the length of the heat exchanger the channels run opposite to each other.

The actual heat transferred between the streams is given by the difference between the heat entering and exiting each channel. For a steady system with no external heat losses from the shell, the heat loss from the primary channel will always be equal to the heat gained by the secondary channel.

$$\dot{m}_1 c_{p,1}\left(T_{1,in} - T_{1,out}\right) = -\dot{m}_2 c_{p,2}\left(T_{2,in} - T_{2,out}\right) \tag{2.8}$$

In order to simplify the parameter testing process, many of the simulations will be carried out with equal flows through each channel. For two fluids of equal heat capacitance and flow rate, through a steady system, the temperature differential across one channel will always be equal to that of the second:

$$\Delta T_A = \Delta T_B \tag{2.9}$$

Applying this relation instead to Equation 2.7, the LMTD becomes indeterminate under these conditions. In this case, the temperature profiles from Figure 2.7 begin and end with equal temperature differentials. LMTD is instead defined as the gap between the two profiles in these scenarios:

$$\Delta T_{LMTD,equal} = \Delta T_A = \Delta T_B \tag{2.10}$$

The LMTD alone is dependent on the scale of the inlet temperatures for each stream of the heat exchangers, and is not by itself a performance metric. The LMTD is used to calculate further, non-dimensional quantities that may be compared across a wider range of conditions. For this reason, the overall heat transfer coefficient, U is calculated. This metric is the proportionality constant for heat transfer, after accounting for the heat flux, surface area and temperature difference. The overall heat transfer coefficient effectively agglomerates together many variables related to the particular fluids, the flow regime and internal shape factors, all of which will realistically be varying throughout the structure, (Shah & Sekulić, 1998). For this reason, studies of the heat transfer coefficient alone must control for many factors, and are often best suited to order of magnitude comparisons. The overall heat transfer coefficient may be determined by Equation 2.11:

$$U = \frac{Q}{A_s \Delta T_{mean}} \tag{2.11}$$

The mean temperature difference ΔT_{mean} represents the difference between the temperatures at the wall and the bulk fluid, averaged along all points within the heat exchange structure. This quantity varies with the velocity distribution within the channels, and any other structural modifications that may be present. Accordingly, ΔT_{mean} is very difficult to calculate, so a correction factor F_t is commonly introduced to Equation 2.12 to compensate for the discrepancy between the true mean temperature difference, and that found by the LMTD.

$$\Delta T_{mean} = \Delta T_{LMTD} F_t \tag{2.12}$$

The LMTD temperature is far easier to calculate, as it is found solely from the inlet and outlet temperatures of the heat exchanger, assuming a linear temperature profile between each entrance. Correction factors for various designs are found by many authors. However, even a single pass shell and tube unit results in complicated expressions (Bhatti *et al.*, 2006). Although the equations that Bhatti et al. formed are trivial to evaluate computationally, the expressions change depending on the heat exchanger type, number of fluid passes, or other structural changes that would impact the flow profile. This makes the correction factor difficult to determine for unusual structures such as TPMS designs and hence the use of F_t for preliminary design would not be possible. Each individual design would require a large amount of testing to determine the correct factor. For this work, it is assumed that the correction value was equal to $F_t = 1.0$, as a more suitable value could not be determined.

Another measure of the mode of heat transfer is the Nusselt number, Nu. This dimensionless quantity is the ratio of convective heat transfer to conductive. Greater values indicate that heat transfer is occurring through the advection and convection of the fluids, through the operation of the heat exchanger, rather than simply conducting through proximity. While the LMTD is often used in the comparison of heat exchanger units, the Nusselt number may be compared across a wide range of applications in different fields, particularly those related to porous media (Kamiuto & Yee, 2005; Teruel & Díaz, 2013), where the LMTD would not typically be associated. The Nusselt number allows for fully dimensionless relationships to be investigates, such as that with respect to the Reynolds number. In addition, the Nusselt number does not require the same geometric information as some metrics, such as effectiveness, as is described later. Equation 2.13 is used to find the Nusselt number:

$$Nu = \frac{hD_H}{k_f} \tag{2.13}$$

The wall heat transfer coefficient h, is calculated from the overall heat transfer coefficient, the wall thickness Δx_w, and the thermal conductivity of the solid wall material k_s. Although each of the hot and cold channels have different coefficient values, the average is used for the calculation of the Nusselt number here for simplicity. For applications where the fluid velocity, and hence Reynolds number is similar within each channel, this value will be similar on each side regardless (Celik *et al.*, 2017). Equation 2.14 gives the wall heat transfer coefficient:

$$h = \frac{1}{\frac{1}{2}\left(\frac{1}{U} - \frac{\Delta x_w}{k_s}\right)} \tag{2.14}$$

The heat transfer coefficient may also include terms for other factors impeding the rate of heat transfer, such as fouling. Fouling however, is very difficult to model accurately, and is often designed around using fixed fouling rates (Nakao *et al.*, 2017). Reasons for this include the fact that the individual particles causing the phenomena are usually on too small of a length scale to accurately model simultaneously with other physics. The action of the foulant particles leaving the bulk fluid, and resting on the heat exchanger wall, to become an interacting part of the geometry is also very difficult. One such study that accomplished a study of fouling was by Kuruneru *et al.* (2019). They used a finite volume, discrete element coupled model to simultaneously track the build-up of foulant particles in the fluid, while measuring the temperature of their metallic foam structure. The present work uses only a single structure, to which making changes over time would be computationally intensive. Fouling may be somewhat mitigated by tightly controlling the medium for which the heat exchanger is used, such as demineralised water.

The value of the characteristic length D_H chosen for the calculation of the Nusselt number varies between applications, depending on the geometry of the system being studied. Systems involving fins may make use of the fin length (Dogonchi & Ganji, 2016). Applications involving beds of spherical particles will often use the particle diameter as the characteristic length (Dolamore, 2017). Porous media studies will use the mean pore diameter (Nakayama *et al.*, 2009). The choice is arbitrary, though the quantities calculated as a result of the chosen length are dependent on its value. For this reason, consistency in the characteristic length is vital between studies.

The number of heat transfer units (NTU) is a metric analogous to the height of equivalent theoretical plates (HETP) and accompanying plate number concepts from mass transfer (Zhang, 2013). The NTU may be compared between designs to determine those with greater heat transfer potential. Although similar to the overall heat transfer coefficient, NTU includes the channel surface area, making it more appropriate in cases where the surface area actually available for heat transfer is uncertain. NTU is given by Equation 2.16:

$$NTU = \frac{UA_s}{c_{p,f}\dot{m}_f} \tag{2.15}$$

The heat transfer effectiveness (Kuppan, 2013) ϵ_u, is the ratio of the heat actually transferred between the two fluid streams, to the maximum that was possible, for a heat exchanger of infinite length. It is defined by Equation 2.17:

$$\epsilon_u = \frac{Q_{Actual}}{Q_{max}} \tag{2.16}$$

Effectiveness is expressed as a ratio on the domain [0,1], where greater values indicate that the heat exchanger unit is closer in transfer ability to one of unlimited length. Since the maximum rate of heat transfer is difficult to determine, the effectiveness is instead linked to the NTU found previously. Equations 2.18 and 2.19 give the effectiveness for parallel-flow and counter-flow configurations respectively.

$$\epsilon_{u,parallel} = \frac{\left(1 - e^{-(1+C_r)NTU}\right)}{1 + C_r} \tag{2.17}$$

$$\epsilon_{u,counter} = \frac{\left(1 - e^{-(1-C_r)NTU}\right)}{\left(1 - C_r e^{-(1-C_r)NTU}\right)} \tag{2.18}$$

The quantity C_r is the mass flow rate weighted ratio of heat capacities for each channel. It is defined such that the numerator is always less than, or equal to the denominator.

$$C_r = \frac{\min\left(\dot{m}_{f,i} c_{p,f,i}\right)}{\max\left(\dot{m}_{f,i} c_{p,f,i}\right)} \tag{2.19}$$

The subscript i in Equation 2.20 denotes the fluid channel. When the two fluids are of the same flow rate, composition, and are close enough in temperature that the heat capacity ratio is approximately equal to unity, Equations 2.18 and 2.19 may be simplified to the following:

$$\epsilon_{u,parallel,equal} = \frac{1}{2}(1 - e^{-2NTU}) \tag{2.20}$$

$$\epsilon_{u,counter,equal} = \frac{NTU}{1 + NTU} \tag{2.21}$$

Heat exchangers are often sized according to the allowable pressure drop for the application (Yang *et al.*, 2020). This makes the pressure drop ΔP, an important quantity for designing full scale exchangers. When comparing the effects of parameter changes on the performance of various designs, the heat transfer coefficient per unit of pressure drop $\frac{h}{\Delta P}$, becomes important. This quantity may be more easily expressed in a non-dimensional manner, as the Nusselt number to pressure drop ratio, $\frac{Nu}{\Delta P}$. Designs with greater heat transfer per unit of pressure drop

are able to transfer more heat for the given pressure conditions in future applications. TPMS designs, while containing many intertwined channels, are not fundamentally different to more traditional designs, such as the shell and tube, whereby the tubes would instead be present in straight bundles. It is assumed that the same heat transfer relations apply.

2.4 Lattice Boltzmann Method

2.4.1 Overview

The lattice Boltzmann method (LBM) is based on the kinetic theory of gases, in which a gas is made up of many small particles, constantly in random motion. These particles are generally imagined as being spherical, and exhibit inelastic collisions between one another. Pressure drops that occur along a surface that a gas is exposed to are due to a reduction in energy incurred by the force of particles pressing against the surface in one location, relative to another. The LBM was used in the current work to simulate fluid flows in the pores, and heat flows in the liquid and solid phases.

The LBM has origins in lattice gas automata (LGA), which is a type of model for gas particles, originally proposed by Hardy *et al.* (1973). The basic premise of an LGA model is that a grid of cells is created, with gas particles moving between. Each site can either contain a particle, or not contain a particle, in a Boolean manner. Collision rules are set out such that whenever two particles reach the same site, at the same time, they are collided away. Particles are tracked individually such that their individual position and velocity at every time step is known. This discretisation of site conditions reduces machine rounding errors for cell particle density. A degree of validation for the computational method was achieved when it was discovered that the Navier-Stokes equations could be derived from a 3D gas automaton (Frisch *et al.*, 1986). The LGA method lost popularity in the 1990s, when the LBM started to be more heavily tested, largely due to the ability to eliminate the statistical noise present in LGA, as developed by McNamara and Zanetti (1988).

The LBM operates in a manner somewhat similar manner to LGA, except instead of tracking individual particles, the LBM uses a probability distribution function. Discrete particles are not tracked; instead fractional particle density is calculated for each node, at each time step, according to the probabilities of particles having travelled there in the previous time step. The Boolean function for the position of each discrete particle is replaced by the probability that the particle went in a certain direction, effectively splitting the particle up. Doing so has the

effect of eliminating statistical noise (Chen & Doolen, 1998). The probability that this distribution represents is both the likelihood of finding a particle in a certain location, in the given system, as well as the average number of particles that would be found in that location, for an infinite number of systems.

As explained by Mohamad (2019), for an arbitrary system containing particles of a gas, the function $f(\boldsymbol{x}, \boldsymbol{u}, t)$ may be used to represent the distribution of the particles within the gas at their respective positions $\boldsymbol{x}$, with velocities $\boldsymbol{u}$, over time t. Emboldened symbols denote vector quantities. After an external force $\boldsymbol{F}$ is applied, without collisions occurring, the number of particles would go unchanged, as new particles would replace those displaced by the external force in Equation 2.23:

$$f(\boldsymbol{x} + \boldsymbol{u}dt, \boldsymbol{u} + \boldsymbol{F}dt, t + dt)\, d\boldsymbol{x}\, d\boldsymbol{u} - f(\boldsymbol{x}, \boldsymbol{u}, t)\, d\boldsymbol{x}\, d\boldsymbol{u} = 0 \tag{2.22}$$

As gases molecules rapidly collide with each other constantly, the external force instead does cause a change to the number of particles present. This difference is represented by the collision operator, Ω:

$$f(\boldsymbol{x} + \boldsymbol{u}dt, \boldsymbol{u} + Fdt, t + dt)d\boldsymbol{x}d\boldsymbol{u} - f(\boldsymbol{x}, \boldsymbol{u}, t)\, d\boldsymbol{x}\, d\boldsymbol{u} = \Omega(f)\, d\boldsymbol{x}\, d\boldsymbol{u}\, dt \tag{2.23}$$

Simplifying down, and taking the limit as $dt \to 0$, gives the result that the change in the distribution function is solely controlled by the particle collisions:

$$\frac{d\boldsymbol{f}}{dt} = \Omega(\boldsymbol{f}) \tag{2.24}$$

Since the particle populations are a function of position, velocity and time, the rate of change of the distribution function can be expanded to become:

$$\frac{d\boldsymbol{f}}{dt} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\frac{d\boldsymbol{x}}{dt} + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\frac{d\boldsymbol{u}}{dt} + \frac{\partial \boldsymbol{f}}{\partial t} \tag{2.25}$$

The quantities $\frac{d\boldsymbol{x}}{dt}$ and $\frac{d\boldsymbol{u}}{dt}$ may be recognised immediately as the velocity $\boldsymbol{u}$, and the acceleration from a force $\boldsymbol{a}$, respectively. The partial derivative $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ is the rate of change of the distribution function with respect to the spatial coordinates of the particles, and may be represented by the quantity $\nabla \boldsymbol{f}$. For a system lacking this external force, and hence acceleration term, the equation may be simplified further into a recognisable advection-diffusion form:

$$\frac{\partial \boldsymbol{f}}{\partial t} + \boldsymbol{u}.\nabla \boldsymbol{f} = \Omega \tag{2.26}$$

When actually solving the Boltzmann equation, Equation 2.24 may be discretised to the following:

$$f_i(\boldsymbol{x} + \boldsymbol{e}_i\delta_t, t + \delta_t) - f_i(\boldsymbol{x}, t) = \delta t\, \Omega(f_i) \tag{2.27}$$

In this form, the Boltzmann equation is not yet readily solvable, due to the complexity of the collision operator. For this reason, significant progress was not made into utilising the LBM, or LGA before this collision term was simplified. This was done in 1954, when Bhatnagar *et al.* (1954) (BGK) formed, and tested their linearised relaxation term. This BGK approximation makes the Boltzmann equation solvable, without introducing large errors.

$$\Omega = -\frac{1}{\tau_v}\left[f_i(\boldsymbol{x}, t) - f_i^{eq}(\boldsymbol{x}, t)\right] \tag{2.28}$$

The choice of lattice depends on the physics being modelled. Each lattice is displayed with the nomenclature D_nQ_m, where D_n is the number of dimensions and Q_m is the number of discrete velocities. In general, the larger the lattice, the more accurate the results will be and the greater the level of information that may be recovered from the moments of the velocity distribution, at the expense of computational time (Saito *et al.*, 2017). The D_2Q_9 is two dimensional, where one central particle has eight neighbours. The D_3Q_{19} lattice is three dimensional, with 18 neighbours. Lattice directions relative to the central node are given weightings according to the Gauss-Hermite quadrature formulae. Examples of these lattices are given in Figure 2.9. While the numbering of each of the lattice directions is arbitrary, consistency with the definitions of the equilibrium distribution functions used is crucial.

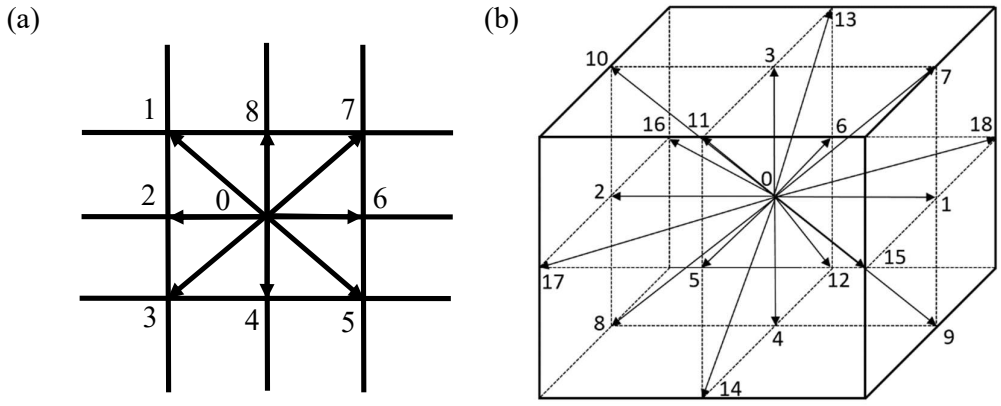


Figure 2.9. Particle arrangements in the a) D_2Q_9 lattice and the b) D_3Q_{19} lattice.

The ability to track the changes in energy was brought about more recently, allowing for the method to be used as a more complete CFD package. Previously, applications were unable to

track energy within the system being modelled, and as such were forced to make the assumption of an isothermal system (Alexander *et al.*, 1993). This greatly limited applications in the past.

Although it has not entered mainstream adoption, the possibility exists for non-uniform lattice grids to be created, such as that described by Peng *et al.* (2004). The feature gives the possibility to greatly enhance the accuracy of results found within regions of large change, without the need to increase the resolution across the entire domain. These non-uniform grids have not had widespread adoption, such that grid uniformity remains one of the greatest limitations of the LBM.

2.4.2 Theory

During each time step of the method, the fictitious particles must move to new nodes on the lattice. Particles are adjusted so that their movements come to approach those of the equilibrium distribution, which may or may not be variable with time. The following theory may be found in any LBM textbook, such as that by Mohamad (2019), but is included here for clarity of the definitions used in this work. The main equation controlling the particle collisions is as follows, for every lattice node:

$$f_i(\boldsymbol{x} + \boldsymbol{e}_i\delta_t, t + \delta_t) - f_i(\boldsymbol{x}, t) = -\frac{1}{\tau_v}\left[f_i(\boldsymbol{x}, t) - f_i^{eq}(\boldsymbol{x}, t)\right] + \delta_t F_i(\boldsymbol{x}, t) \tag{2.29}$$

where i is the lattice direction integer, on the interval $[0, m-1]$ for a D_nQ_m lattice, and f_i is the population density in each direction. The vector quantity $\boldsymbol{x}$ gives the cartesian coordinates of the node at which the equation is evaluated, while t is the current time step. The quantity $\boldsymbol{e}_i$ is the lattice unit velocity for the direction i, $\delta_t = 1$ is the dimensionless time step, τ_v is the relaxation time and F_i is a body force term (Peng *et al.*, 2004).

LBM requires an equilibrium distribution function, f_i^{eq} to exist, towards which the relaxation parameter will always push the system. The inclusion of the relaxation parameter comes about due to the statistical nature of the method, and the gradual movement towards a constant state through the influence of entropy. The equilibrium distribution function f_i^{eq} is given by:

$$f_i^{eq} = \rho w_i\left(1 + \frac{\boldsymbol{e_i}.\boldsymbol{u}}{c_s^2} + \frac{1}{2c_s^4}(\boldsymbol{e_i}.\boldsymbol{u})^2 - c_s^2|\boldsymbol{u}|^2\right) \tag{2.30}$$

where w_i is a set of constants providing directional weighting dependent on the lattice chosen, $\boldsymbol{e}_i$ is the directional unit vector, c_s is the lattice speed of sound, $\boldsymbol{u}$ is the physical velocity and

ρ is the physical density at the current node. The streaming step is essentially the final piece of the algorithm, at which point the time step is incremented, and the particle distributions are moved forwards onto the next adjacent nodes in each of the Q_m directions. The streaming equation is as follows (Mohamad, 2019):

$$f_i^{in}(\boldsymbol{x} + \boldsymbol{e}_i \delta t, t + \delta t) = f_i^{out}(\boldsymbol{x}, t) \tag{2.31}$$

Most quantities may be found by taking moments of the population distribution function. This is achieved by weighting the distribution function, before carrying out a summation of all directions at the specified node, to calculate the moment.

The particle population distribution function may be represented by f_i where for a $D_n Q_m$ lattice, i may take values from 0, the centrum, up to a limit of its $m - 1$ neighbours. From this, the zeroth order moment, the mass density, may be computed by the following equation:

$$\rho = \sum_i f_i \tag{2.32}$$

The velocity in each of the Cartesian directions may be computed by a similar formula:

$$\boldsymbol{u} = \frac{1}{\rho} \sum_i f_i \boldsymbol{e}_i \tag{2.33}$$

The velocity magnitude across three dimensional space may be simply determined through Pythagoras' Theorem for vector addition:

$$|u| = \sqrt{u_x^2 + u_y^2 + u_z^2} \tag{2.34}$$

where u_x, u_y and u_z are the three cartesian components of the physical velocity at any point. For the purposes of determining the pressure drops across the entrances, the physical pressure P may be found from the density by Equation 2.36 (Feng *et al.*, 2007; He *et al.*, 1998). The physical density ρ_{phys} is the density of the fluid at the average temperature of the simulation, in physical units.

$$P = \rho c_s^2 \rho_{phys} \frac{dx^2}{dt^2} \tag{2.35}$$

When a simulation cannot be assumed to be isothermal, temperature effects must also be considered. Inclusion of the energy conservation equations is not trivial, though there are multiple methods now to achieve this. Older methods (Alexander *et al.*, 1993) keep the same, single distribution function and compute higher order moments to find quantities such as the

heat flux and total energy. These typically require lattices with larger numbers of neighbours for sufficient information to be available. Due to every nodal point on the lattice structure having more neighbours, the computational speed is slowed. Since all data is stored together in a single lattice, this method can be significantly inhibited by memory requirements, potentially more so than the alternatives.

One alternative is the double distribution method (Peng *et al.*, 2004), whereby a second distribution function and accompanying lattice are created for tracking the energy. Within this energy lattice, the result of the zeroth order moment yields the temperature, instead of the density, as in the case of the hydrodynamic lattice. The collision equation for the energy lattice is identical in form to previously:

$$\begin{aligned} &g_i(\boldsymbol{x} + \boldsymbol{e}_i\delta_t, t + \delta_t) - g_i(\boldsymbol{x}, t) \\ &= -\frac{1}{\tau_g}\left[g_i(\boldsymbol{x}, t) - g_i^{eq}(\boldsymbol{x}, t)\right] + \left(1 - \frac{1}{2\tau_g}\right) Q_i(\boldsymbol{x}, t) \end{aligned} \tag{2.36}$$

The quantity Q_i is a heating source term, used optionally for non-boundary heat additions, heat creation due to reactions, and some temperature correction schemes. The coupling is implemented through copying the velocity terms from the hydrodynamic lattice, while the temperature of the energy lattice is used for a body force correction of the density. While this method is relatively simple in execution, the use of a second lattice can cause stability problems due to the difference in the key diffusivity term of each (Perko & Patel, 2014).

The equilibrium distribution mirrors that of the hydrodynamic lattice. The velocity terms are copied directly, while the temperature T replaces the equivalent thermal lattice density.

$$g_i^{eq} = Tw_i\left(1 + \frac{\boldsymbol{e_i}.\boldsymbol{u}}{c_s^2} + \frac{1}{2c_s^4}(\boldsymbol{e_i}.\boldsymbol{u})^2 - c_s^2|\boldsymbol{u}|^2\right) \tag{2.37}$$

Another alternative is to instead couple a finite difference method (FDM) for use in the calculation of the temperature (Lallemand & Luo, 2003). Through this approach, the LBM is used to calculate the velocity field, while finite differencing is used to track the temperature. Rather than the LBM particle density approach to tracking the flow of heat, more traditional advection-diffusion laws may be used in the bulk fluid, while Fourier's Law is utilised for the solid. What this method loses in speed compared with the double distribution method, it gains in stability from only utilising a single lattice. The FDM approach has two variants. The finite differencing scheme may be used for the entirety of the thermal calculations, or it may be reserved only for the solid phase. The solid phase, or any other area of heat capacitance

differing to the bulk fluid, is the primary region for which instability traditionally arises within thermal LBM. Limiting the application of the FDM to only this most troublesome region preserves as much of the efficiency of the base LBM as possible.

In order to accurately compute these quantities, a 3D lattice containing at least 19 directions per node is required, for example the D_3Q_{19} lattice. When the double distribution method is instead utilised, the temperature and heat flux are calculated by the following:

$$T = \sum_i g_i \tag{2.38}$$

$$\boldsymbol{q} = \left(1 - \frac{1}{2\tau_g}\right) \sum_i g_i \boldsymbol{e}_i \tag{2.39}$$

Compared with the older methods using only a single hydrodynamic lattice, the double distribution function method results in reduced computational load and greater numerical stability, as less information in the form of higher-order velocity terms is required (He *et al.*, 1998; McNamara *et al.*, 1995). In this case, the thermal lattice may be lowered in size to the D_3Q_9 lattice, giving large potential performance improvements, at the cost of some accuracy. In practice, while the double distribution method is used in the present work, the thermal lattice is maintained at the same size as the hydrodynamic. This is done for the aforementioned accuracy improvement, as well as algorithmic simplicity when incorporating additional functionalities. A comparison of the effects of these different energy lattices was beyond the scope of this work.

2.4.3 Applications of the LBM

Although the lattice Boltzmann method has now been used in a wide variety of applications, a few interesting implementations are listed here, where certain features were included, challenging the limitations of the algorithm. The LBM is shown to be usable across a range of conditions, and for a variety of applications.

The LBM performs well when being used for modelling low Reynolds number applications, where a low simulation resolution may be used, and hence be computationally fast (Mohamad, 2019). In a study by van der Hoef *et al.* (2005), the LBM was used to simulate fluid movements with Reynolds numbers below 0.2, passing through a particle bed of spheres. From this, they were able to match the drag forces on the particles within the bed with high precision to a

modified version of the Carman equation. As they operate at very low Reynolds number, microfluidics applications present optimal conditions for the use of the LBM. Montessori *et al.* (2018) demonstrated a regularised multicomponent LBM formulation for purpose of simulating individual droplets forming within experimental microfluidics apparatus, moving with a Reynolds number of 10. They were able to correctly predict the size and behaviour of individual droplets extruded from the nozzle of a microfluidics device, matching closely to experimental data. Each of these studies demonstrate a high level of accuracy between numerical and experimental results and existing low Reynolds number theory. Many more microfluidics applications were summarised by Zhang (2011). This review encompasses applications across heat transfer, electromagnetism, diffusion, single and multiphase flows, analysis of interfacial dynamics, biological processes and porous media where microfluids were accurately simulated.

While the LBM is more commonly used for low Reynolds number applications, examples also exist for higher Reynolds number cases. In a study comparing the effects of various boundary conditions on a standard 2D lid-driven cavity benchmark case, Hu *et al.* (2017) were able to investigate Reynolds numbers up to 7500. It was found that for their simple geometry, the specific type of boundary condition used did not make a significant difference. In another study by Di Ilio *et al.* (2018), a semi-Lagrangian approach was taken for 3D turbulence modelling of fluid passing a cylinder at a Reynolds number of 3900. They made use of a non-uniform grid, allowing smaller spacing, and hence higher resolutions in areas likely to cause instability, greatly increasing stability. A much wider spacing, and lower resolution was possible in open areas away from the walls, saving on computational costs.

Another study demonstrated the potential for heat transfer (He *et al.*, 1998). In this work, an energy distribution function independent of that for the mass was implemented, allowing for temperature to be modelled, without introducing instability to the simulation, or requiring an extremely limited range of thermal or flow conditions (Alexander *et al.*, 1993). This was the work where the concept of the double distribution function was formed, whereby one lattice tracks the fluid movements, while a second lattice controls the flow of heat.

A two-dimensional thermal LBM simulation, commonly used as the basis of a benchmark based on a Rayleigh–Bénard convection problem was demonstrated by He *et al.* (1998). Here, a layer of viscous fluid in a box was heated from beneath, while the top surface is maintained at another, known temperature. Since the system can be solved analytically, it is useful for verifying simple, 2D thermal techniques. In the case of thermal flows, a few of the common

benchmarks used to validate LBM results are based on Rayleigh-Bénard convection for fluids, while regions of differing thermal conductance are used for solids. Since the fluids are at times static, and involve natural convection, the Rayleigh number is used for comparison instead of the Reynolds number. Thermal benchmark tests are generally separated into steady state and dynamic categories. While adjustments are commonly required to thermal LBM to allow for accurate transient behaviours to be modelled, even dynamically incorrect methods will usually yield the correct steady state solution. Imani (2017) demonstrated a method to apply a correction to the lattice Boltzmann particle streaming step to allow for dynamically accurate simulation of a simple natural convection box benchmark case. The work showed close alignment to the standard results of the benchmark. This study, by Imani in 2017, provided the first dynamically accurate conjugate heat transfer simulations for the three-dimensional LBM, in situations without uniform heat capacitance.

2.4.4 Alternative CFD methods

Although most LBM implementations make use of custom written, open-source programs, alternative CFD methods often utilise commercial packages. One example is the program Fluent, which makes use of the finite volume method. In the finite volume method, conservatory partial differential equations are converted into a series of linear algebraic equations, which may be solved (Moukalled *et al.*, 2016). The simulation space is broken down into a series of control volumes, at the centre of which is the integration point. Interpolation is used to calculate the values of the simulation variables at each of the control volume surfaces (Tu *et al.*, 2012). The finite volume method is readily usable in high Reynolds number scenarios, and can easily make use of unstructured meshes for more stable simulations. The commercial CFD program Fluent has become one of the most widely used CFD programs in academia. It is well supported, and makes the initial setup phases of the modelling more straightforward. One popular open-source alternative for a finite volume method CFD program is OpenFOAM. OpenFOAM is free, and while lacking in the level of usability of Fluent, OpenFOAM has greater utility, through a large number of community written plugins that may be incorporated into the base numerical solvers, to aid in various forms of physics.

A comparison of OpenFOAM and Fluent was made by Ariza *et al.* (2018) for the purposes of a 3D, high Reynolds number, steady state annular reactor. The results of the simulations were compared with experimental data to verify the results. They found the accuracy of the end results very similar for each program tested and while Fluent was at times easier to use due to

the well developed graphical user interface, OpenFOAM benefits from the customisability and lack of cost of open-source code.

Another alternative CFD method is the finite element method. In the finite element method, surfaces are split up into discrete elements, upon which a conservation equation must be solved for each variable. These equations are then solved together, with the solution to the system satisfying a given error function. On complex surfaces, where many elements are required, potentially many millions of equations may require solving (Mayboudi, 2020). The finite element method can also have stability issues and it is computationally slower than many of the alternatives. COMSOL is an example of a commercial CFD program making use of the finite element method (Mayboudi, 2020), with optional coupled solvers making use of various other methods. With a useful graphical user interface, COMSOL can be easier to use when mesh geometries are simple, often either drawn locally within the program, or imported from an external computer aided design program such as SOLIDWORKS. However, the procedurally generated meshes required for the present work, as formed in Chapter 3, do not guarantee such straight edges. While the interior regions of the design are well defined, the edges can be jagged. In addition, the need to assign boundary conditions to every surface would make COMSOL difficult to work with when dealing with complex TPMS structures.

The finite difference method is another commonly used CFD technique. In finite differencing, numerical derivatives are calculated for the differential equations being studied on each individual nodal point on a grid. The grid need not be uniformly spaced, but limits exist for the amount of grid distortion before accuracy is affected to an unacceptable degree (Tu *et al.*, 2012). When writing a custom made CFD program, the finite differencing method is one of the least troublesome to compose, being taught widely throughout undergraduate courses. However, boundaries can be difficult to set on the edges of a surface and the computational efficiency of the method can be low. Regardless, the finite difference method is a useful tool when initially investigating a CFD problem.

3 Creation of printable structures

3.1 Introduction

As discussed in Chapter 2, the most widely used file type for 3D printing is the STL file. At its most basic form, the STL file is simply a collection of triangles made up from their constituent vertices: points in space connected together through numbered faces. Each face has a normal direction to aid in the translation between void or solid. Ideally, every edge of every triangle within the file will have exactly two face neighbours. These triangles form a surface, which when wrapped around in three dimensions, outlines a solid body. Upon being sent to the printer, a voxelisation algorithm determines the dimensions of the solid body based on the assumption that the outermost material must be void.

An object is said to be "manifold" if for every triangle within it, every edge is connected to exactly two faces. This means any edge containing an intersection between more than two surfaces, or an edge with only one face attached, is not valid. Depending on the voxelisation algorithm deployed, such cases may make a structure unsuitable for either printing or simulation. As the bulk of the packing domain is governed by the TPMS equation in use, the most likely location for non-manifold edges are the domain boundaries.

The creation of printable TPMS is a developing field in additive manufacturing. Although the TPMS equations provide the necessary information to generate a surface from which a unit cell of a design may be generated, using this information to create a full, multi-cell structure was a challenge at the beginning of the present work. In many early works, such as that by Kadkhodapour *et al.* (2015) a single unit cell was formed, and then replicated in space using a computer aided design (CAD) program. Such methods are computationally expensive, and can encounter STL file resolution issues. A subsequent work by some of the same authors (Afshar *et al.*, 2016) resulted in an image based method, where a single unit cell was voxelised, and the resulting 2D images instead replicated and sent to the printer. This method is far more computationally efficient, though would appear to make altering the structure more difficult. More recent studies, such as that by Feng *et al.* (2019) focus on methods of algorithmically generating instructions for 3D printing TPMS structures without storing large data structures in memory. This method allows for the construction of very large parts, bypassing the usual memory limitations of current 3D printers.

3.2 Method

The most fundamental part of any simulation work is the domain and boundaries on which the simulation is run. These must be clearly defined, without room for reinterpretation. For the present work, a TPMS packing contained within a structural casing was desired. The TPMS packing needed to be generated to a high degree of precision, without errors, while the simpler shell required mainly to just be of the correct dimensions. For this reason, the generation of the simulation domain was split into two parts: the creation of the packing from the base equations in MATLAB, and the build of the shell in SOLIDWORKS.

At its core, the generation of a TPMS is a root finding problem. The trigonometric equations for these surfaces are too complicated to have three dimensional coordinates explicitly calculated for them. Instead, implicit methods must be employed to approximate the locations of the roots of the equation, which in turn is taken to be a location on the surface.

The first attempt at generating TPMS structures was through the use of a Newton's Method script in MATLAB. In theory, after solving a TPMS equation, the same points at which it was solved could be used as vertices in an STL file. A simple script was written where a multidimensional Newton's Method was employed to implicitly solve the TPMS equations. For the example of the gyroid,

$$f = \sin(ky)\cos(kx) + \sin(kz)\cos(ky) + \sin(kx)\cos(kz) - t \tag{3.1}$$

the equation was solved for z, with fixed $f = 0$, and x and y as uniform 2D arrays of coordinates. The parameter k is the periodicity, which becomes an arbitrary parameter for STL file generation, since the resultant vertices are usually scaled to fit within the bounds of a desired domain at the end. The t term is an offset, the uses for which will be expanded upon later. This method, while conceptually simple, suffered from instability in the numerical solutions produced in regions with large gradients $\frac{dz}{dx}$ and $\frac{dz}{dy}$. As such, z coordinates could be generated for the surface accurately only on isolated slices of the z axis from the domain, but not for areas where these slices linked together. Figure 3.1 shows a typical result of such an approach.

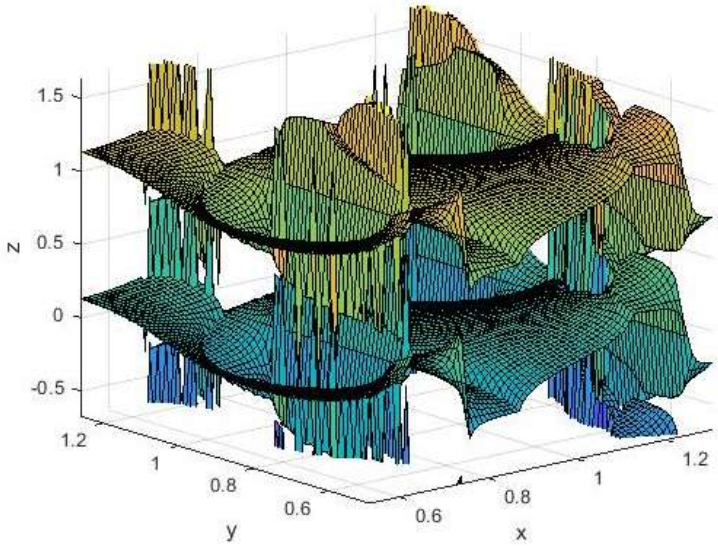


Figure 3.1. Newton's Method generation of TPMS geometries.

There were a number of reasons for the direct calculation method not working. One issue was due to the periodic nature of the TPMS equations, solutions for any fixed cell could potentially diverge to a different cell, depending on the initial guess. One possible solution to this problem is to place bounds on the acceptable range for z coordinates calculated, such that solutions may be limited to only a single unit cell. From this single unit cell, additional cells could be calculated, due to the property of all cells being identical. Although the bounds placed on the solutions would prevent the divergent behaviour from occurring and would reset the guess with reduced step size, doing so would make the solutions difficult to find in some locations. Another issue was the presence of more than one z solution within each unit cell for most combinations of x and y coordinates. Although this too could be overcome with domain bounds and step size reductions, through splitting the unit cell into four quadrants calculated separately, the solution method lacked scalability. Different TPMS, each with different cuts and orientations to be created on meant there was no single method for automating their creation.

An alternative method was devised whereby the domain data was generated first, before finding the actual surface points in a second step. First, as in the earlier attempt, the location data for a uniform grid was first formed over the domain of interest. The function value at each of these points was calculated, with any change between positive and negative value being of interest. The inbuilt MATLAB *isosurface* function was used to interpolate this 3D grid of function values, in order to find the locations where vertices could be situated. This function makes use

of the marching cubes algorithm created by Lorensen and Cline (1987). These locations were where the function values were calculated to be equal to $f = 0$. Although various polynomials were trialled, no significant differences were found compared with simple linear interpolation. The function also grouped these vertices together for ready translation into the STL file format later. This process, while fast and accurate, is only capable of separating the domain into two regions. Therefore, the only usable geometries are single channel in nature, with exactly one fluid and one solid phase. An additional geometry, of two fluid channels separated by a wall of zero thickness may also be formed, though this scenario is limited in use, due to its non-physicality. Figure 3.2 demonstrates such as structure. While these are useful for some applications, such as generating entirely solid basic geometric shapes, an extra level of detail is required for a heat exchanger wall.

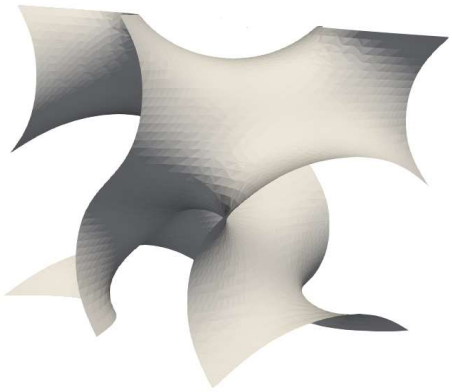


Figure 3.2. Single channel TPMS generated.

To give the heat exchanger walls thickness, a second surface, at a pre-specified offset was also generated. For TPMS equations, the final term is always the offset. In the example of the gyroid given by Equation 3.1, the t term is proportional to half of the wall thickness. At this point, two surfaces are present, parallel across all space, with an empty gap between them. For this gap to be registered as a solid during either printing or simulation, it must be fully enclosed on all sides. Each edge of the packing domain must therefore be altered by way of changing the sign on the function value, in order to force triangles to be formed between the two surfaces, as shown in Figure 3.3. The choice of sign on the outer edges determines which set of channels are open, and which set are blocked off. An incorrect choice results in adjacent inlets and exits being open to each other, effectively shorting the heat exchanger. Each of the unit cell designs used are shown in Appendix A.

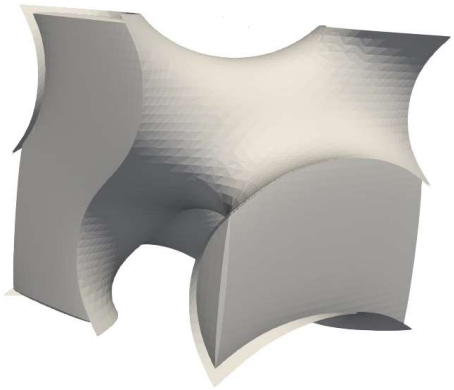


Figure 3.3. Edges of STL for channel sealing.

Once the packing isosurface has been formed, a shell is needed to contain it. Shells are generated in SOLIDWORKS, according to the dimensions of the already formed packing. If the coordinates of the build within SOLIDWORKS are chosen to be the same as those used within MATLAB, later lateral translation of one of the sets of vertices can be avoided. The heat exchanger design tested utilizes two inlets and two outlets, positioned perpendicular to each other. The entrances perpendicular to the main axis of the heat exchanger include flow distributor channels that run around the exterior, allowing easier fluid ingress to the packing.

The design of the shell follows that of Symon (2017). One channel each exists for the hot and cold streams, arranged in a counter current arrangement perpendicular to each other. The streams approaching perpendicularly utilise flow distributors around the outer wall to ensure a free path exists to the interior of the heat exchanger channels. Figure 3.4 shows a cross-section containing a gyroid design. Appendix B gives the shell view for other interiors tested.

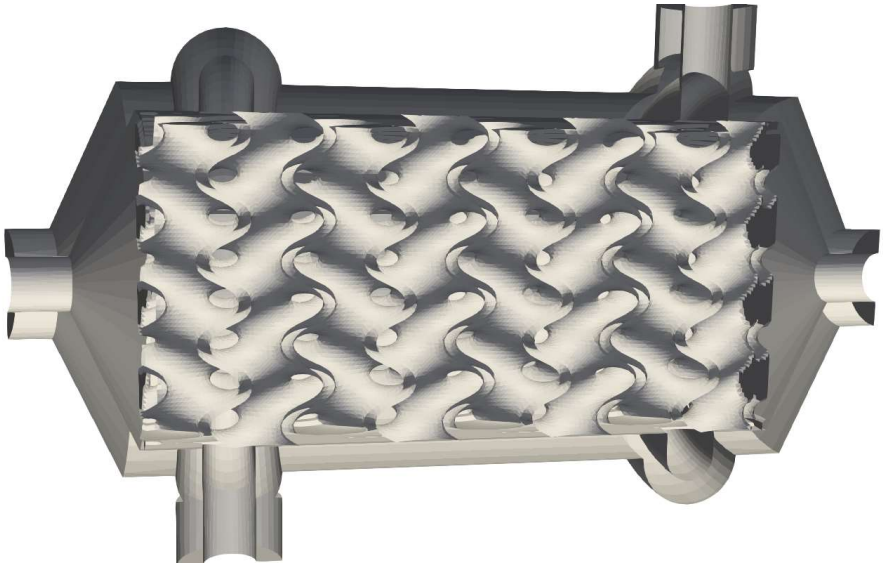


Figure 3.4. Cross-sectional view of the full-scale heat exchanger design.

Once the SOLIDWORKS part has been formed, it must be exported as an STL to be capable of being used in conjunction with the rest of the geometry. Although SOLIDWORKS has built in support for the writing of STL files, this functionality was found to be unreliable when working with parts containing fine details. Additionally, the file sizes generated through this method were often much larger than through other methods. For these reasons, SOLIDWORKS was only used to generate the shell, for which a much lower resolution was permissible.

With the entirety of the walling now enclosed by triangles and the shell read back in, the final step of the process was to splice all of the surfaces together and write them all as a single STL file for export. To do this, all of the vertices and faces from each surface were concatenated together, before being sent into a modified version of the MATLAB *stlWrite* function. This function simply writes out to a file all of the previously found faces and vertices, in the standard STL file format. The standard version of this function was required to be altered in the present work to allow for vectorised processing of the STL files. Without doing so, only a single STL may be processed at a time, and multiple surfaces cannot be combined. This step can be checked to have worked correctly if the ASCII version of the file is selected, allowing it to be human readable. The alternative binary version is more space efficient, so is more commonly produced. Checks were run such that adjacent triangles could only deviate in angle by a fixed tolerance, helping ensure the smoothness of the mesh. The resultant file may then be additionally checked through programs such as MeshLab and Blender to ensure that the surfaces are manifold. These programs have built-in hole detection and hole-mending algorithms, as well as the capacity to completely rewrite the mesh at a different resolution.

3.3 Results and Discussion

The resolution of the geometry, N_{mesh} is an important factor to its viability. During generation, the resolution is defined here as the number of points in each cartesian direction for which the function is evaluated, within each unit cell of the geometry. This number is proportional to the final number of vertices created for the STL. The periodicity, k is the number of function repetitions per unit of the spatial domain. When the periodicity is increased, the number of calculation points remains the same, while increasing the number of periods these points must be shared across. Doubling the periodicity will have the result of twice as many units of the specified geometry becoming present within the same previous physical space. This has the effect of decreasing the effective resolution, N_{eff} of the file through the equation:

$$N_{mesh} = kN_{eff} \tag{3.2}$$

When the effective resolution is too low, the chances of errors occurring in the meshing process are increased. Triangles may be formed intersecting with each other, non-manifold edges may be created, or the generated geometries may no longer be accurate depictions of the desired TPMS. In Figure 3.5, the low resolution image marks the limit below which the resolution cannot be lowered further without significant errors occurring. At this point, sharper edges are just starting to form, and further reductions in resolution will result in partial blockages of the TPMS channels.

Figure 3.5. Comparison between resolutions, N_{mesh}, for STL generation. A) $N = 10$. B) $N = 30$.

Increasing the resolution can prevent the aforementioned issues, but comes at the costs of file size and calculation time. With an effective resolution of 30, the mesh no longer contains abnormalities, however the file size increases at an approximately cubic rate, while the generation time is much longer again. This trend continues until the point where all of the primary memory of the machine has been allocated, and paging to the hard disk occurs, massively slowing the process. It should be noted that the geometric resolutions used within this chapter for the formation of STL files are separate to the numerical resolutions used for simulations in further chapters. The exception to this is if an STL file of high resolution contains details which may be lost when revoxelising back into a relatively lower resolution simulation space.

Once the STL file for the part was obtained, the heat exchanger could then be checked within the Blender program to ensure that the geometry was manifold, and that the boundaries had the correct orientations. The file could then be sent to one of the 3D printers within the University

for printing. Figure 3.6 shows a cross-section of one such design. Doing so in plastic first allowed for any issues in the part to be identified quickly and cheaply.

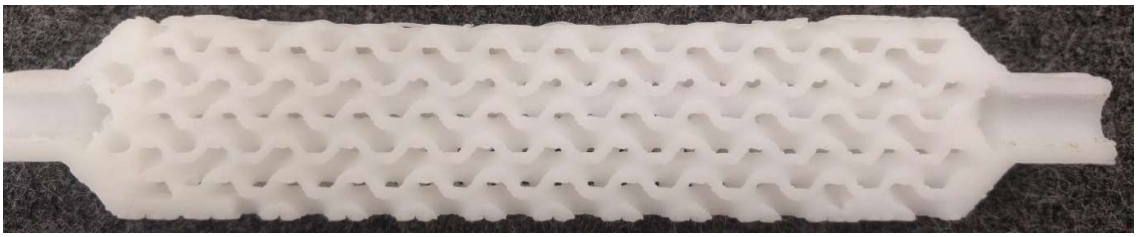


Figure 3.6. Cross-section of a polymer TPMS 3D printed heat exchanger.

The final parts were required to be printable in metal, for a realistic comparison to a heat exchanger to be possible. To do so, alterations were required for the design. Nicholas Symon (Symon, 2017) created the stainless steel TPMS heat exchangers in Figure 3.7, while the present work aimed to more simply replicate the design process, through the previously detailed use of MATLAB. In summary of his findings, to prevent the entrapment of metallic particles within the heat exchanger channels, most of the outer shell was removed. Since the local 3D printers were unable to print with metals, the part was sent to an external business, Ram 3D, for printing by the selective laser sintering method. Images of additional SLA printed prototype parts are included in Appendix C.

Figure 3.7. TPMS heat exchangers printed from stainless steel.

Since the TPMS equations are highly non-linear, the interpolation algorithm used to place the vertices of the isosurfaces can develop some amount of error. This error should be minimized,

as it represents a deviation from the specific shape desired. The accuracy of the vertex placement with respect to the original equation used was checked by recalculating the function value at each, and finding the resultant average error for each vertex. The value of this error was found to be negligible.

The MATLAB program is also capable of generating other geometries. For each other geometry, depending on their use, a shell is required to be generated separately in SOLIDWORKS. By default, cuboid unit cells of TPMS structures may be constructed without alterations. Such geometries are useful when only unit cells of TPMS are desired, and other design considerations are to be ignored. Shapes other than cylinders may also be cut from the domain. By inputting the equation for a sphere, a TPMS structure shaped to an ellipsoid sphere may be formed, as in Figure 3.8. Although the process of generating other custom shapes is in theory as simple as altering the equation to which the geometry is cut, considerations of the wall thickness must always be made. For steeply curved geometries, a simple linear offset between the two surfaces may be insufficient to provide a constant wall width. In this situation, one surface may require a deformation to ensure a proper separation distance.

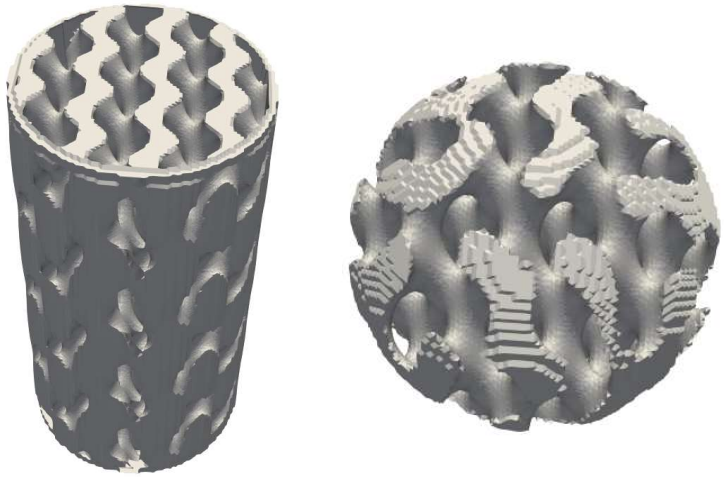


Figure 3.8. TPMS generated from arbitrary equations. A) Cylinder. B) Sphere.

The domains for which the TPMS structures are generated need not be defined by geometric equations. Figure 3.9 demonstrates the result of taking an existing STL file of a closed surface, parsing it into the program, voxelising, and using as the domain for TPMS generation. By taking the STL form of an existing TPMS shape, pseudo fractalline designs may be formed. Fractals are patterns that exist along all length scales, infinitely repeating within themselves. By embedding such a gyroid structure within the walls of another greater gyroid, the beginnings of such a design may be formed. Such a structure could be desirable in applications of porous

media, such as chromatography, whereby TPMS are desired both for the macroscopic flow channels throughout the packed column, and for the interior of pores through which adsorption or other chemical process could occur. In this manner, any complex geometry may be filled with a TPMS structure.

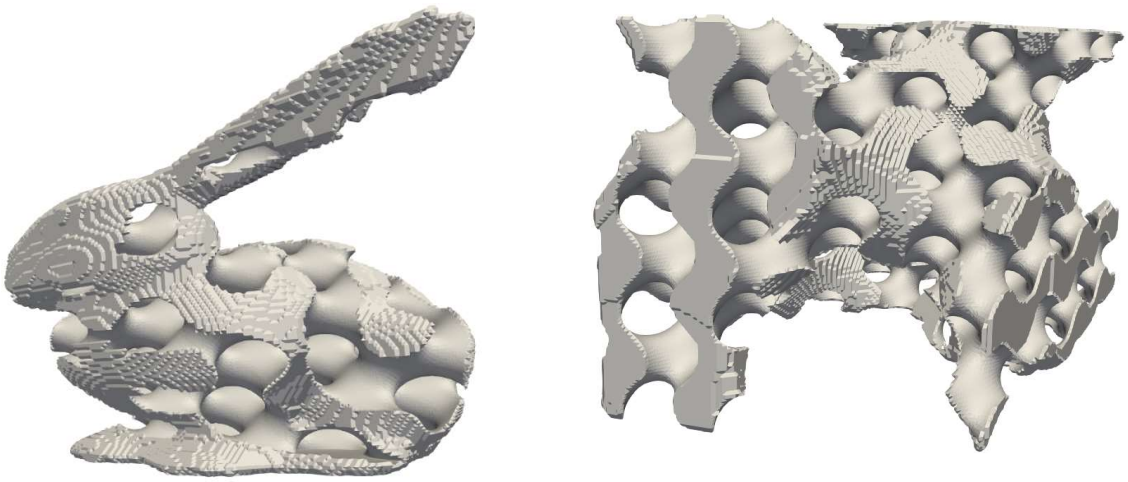


Figure 3.9. TPMS generated from existing STL files. A) Rabbit. B) Fractalline gyroid.

3.4 Conclusions

A MATLAB program was developed through which the generation of STL files from equations was possible. Although the geometry must exist on the three dimensional real domain, the equation may be in either an implicit or an explicit form. The bounds for these geometries may be supplied either in equation form, or from another STL file, which may be revoxelised.

The code generates files which may be passed to a 3D printer for manufacturing and testing, or to a simulation program for more in depth analysis. Each of the TPMS parts generated may be either single channelled, or sheet in nature.

Post-processing changes in external, open-source programs such as Blender are necessary to ensure the integrity of the STL files produced for use in 3D printers. However, this same level of rigour is not required for simulation. The two applications largely differ in the final voxelisation algorithms used, whereby great precision may be found in the shape generation step, while a level of course granularity remains in the simulations. Further differences may also occur due to the individual printer used for the physical printing of final designs.

4 Numerical Methods

4.1 Overview

A number of different computational models were constructed in attempts to accurately study the TPMS heat exchanger system created in the previous chapter. The first version was a MATLAB code whereby a 3D unsteady PDE system was solved through the method of lines. Next, a C++ LBM code was constructed, whereby an STL file of the entire heat exchanger part could be input by a user, from which velocity profiles could be determined. To improve the stability of this system, and allow the calculation of thermal effects, the third method, a representative elemental volume (REV) approach using much of the same code was built.

Although the MATLAB version was comparatively simple to produce, many more assumptions were required than other methods, in part due to the functional approach of the programming language itself. It was deemed too difficult to simulate specific, complex 3D structures, especially when multiple materials were to be involved. In addition to this, a computational limit was reached on the speed of execution, while the setup of parallel processing for “non-embarrassingly parallel” situations was deemed non-trivial.

The LBM models, although sharing a common language, and method of application, are useful in different regards. The full-scale approach allows for the calculation of flow profiles over the entire geometry, as well as allowing the possibility of examining entrance effects, and other phenomena created from the outer walls. The method is lacking in numerical stability while calculating thermal effects, largely due to the value of the thermal diffusivities, which are similar to, or orders of magnitude higher than the kinematic viscosity. Depending on the materials in use, the effect may be made worse from the use of highly conductive metals. The REV approach greatly increases this numerical stability, at the cost of requiring more assumptions into the uniformity of the base TPMS unit cells making up the heat exchanger tubes. Doing so limits the range of applicability of the method, but greatly reduces computation time.

4.2 Finite Difference

A finite difference MATLAB model is first developed as preliminary work, with the aim of investigating only the physics, and disregarding the geometry. As such, this particular model makes the assumption of a uniform packing, without any considerations of shell effects. In this

version, two fluid channels exist, separated by an arbitrary geometry. The packing is assumed to exist uniformly across the domain. Heat transfer coefficients and areas are taken from experimental data, and are the only difference between different configurations.

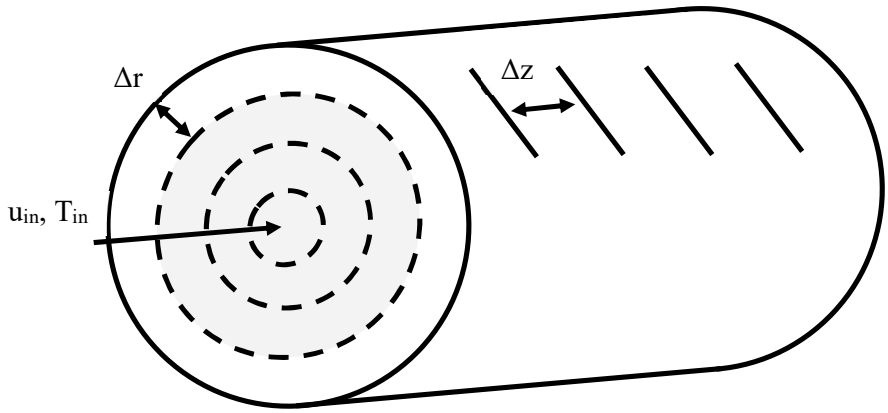


Figure 4.1. System modelled in finite difference simulations.

The finite difference model in Figure 4.1 is 3D, using an assumption of radial uniformity, and temporally transient. Within the cylindrical coordinate system, all radial segments are spaced evenly, such that although the radial distance is always the same. The innermost segment has the smallest area, and hence volume, while the outermost segment has the largest of each. Segment zero exists on the centre line of the cylinder formed, causing it to have zero area and volume. All axial segments are evenly spaced, causing all to be of the same size. Equation 4.1 gives the volume of any particular segment along the radius of the cylinder, where the z axis is the axial direction, which Δz is a segment of.

$$V_j = \frac{\pi\left(D_{j+1}^2 - D_j^2\right)}{4}\Delta z \tag{4.1}$$

The areas available for heat and mass transfer between segments differ depending on the direction of the transfer. In the axial direction, the area of the fluid is that of an annulus, using Equation 4.2, based on the definition of volume formed in Equation 4.1. In the radial direction, the area is that of the outer walls of a cylinder with a diameter equal to that of the present segment. These areas are given by Equations 4.2 and 4.3 for the fluids, subscripted as f.

$$A_{ax,f,ij} = \frac{1}{2}\frac{V_j}{\Delta z}\phi \tag{4.2}$$

$$A_{r,f,ij} = \frac{1}{2}\pi D_j \Delta z \phi \tag{4.3}$$

Each of these quantities is constant in the axial direction, while variable only in the radial. The subscript i is used to denote the axial direction, up to a limit of m. The subscript j is used to denote the radial direction, up to a limit of n segments. Since the volume V is calculated under the assumption of an empty tube, a porosity term ϕ, must be incorporated to account for the proportion of each cell is taken up by the fluid. The remainder of the tube, $1-\phi$, is the solid. Each of the fluid terms is halved to account for the inclusion of two distinct fluid channels. In the case of thermal movements through the solid, subscripted as s, these areas are modified to:

$$A_{ax,s,ij} = \frac{V_j}{\Delta z}(1-\phi) \tag{4.4}$$

$$A_{r,s,ij} = \pi D_j \Delta z(1-\phi) \tag{4.5}$$

The temperature of the fluid, T_f is affected by the advection of fresh fluid into the channels, diffusion of heat within the fluid, and the rate of heat transfer through the walls. Like those of the fluid, each of the packing segments, from whence the heat transfer area is derived, are of unequal size. As such, to simplify the model, it is assumed that every segment has a share of the total wall surface area proportional to its volume. Although this is physically unrealistic, the simplification is made for cases where many unit cells of the desired TPMS structure are present, and simulation segments are large relative to the diameter of the channels. The heat transfer area per unit, $A_{p,j}$ is equal to the total surface area of the packing A_p, multiplied by the fractional volume of the total taken up by the segment in question.

$$A_{p,j} = A_p \frac{V_j}{V} \tag{4.6}$$

In calculating the temperature of the fluid and solid, each of the terms are calculated separately in the axial and radial directions, for each of the transfer components. These components are the advection of fresh fluid, the conduction through like media, and convection through the walls. This process is then repeated for each of the two fluids, and the solid walls.

$$\frac{dT_i}{dt} = \left(\frac{dT_i}{dt}\right)_{ad} + \left(\frac{dT_i}{dt}\right)_{conv} + \left(\frac{dT_i}{dt}\right)_{cond} \tag{4.7}$$

$$\frac{dT_j}{dt} = \left(\frac{dT_j}{dt}\right)_{ad} + \left(\frac{dT_j}{dt}\right)_{conv} + \left(\frac{dT_j}{dt}\right)_{cond} \tag{4.8}$$

$$\frac{dT}{dt} = \frac{dT_i}{dt} + \frac{dT_j}{dt} \tag{4.9}$$

The primary means of thermal movements within each channel is the advection of fluid through it. In this version, the density and velocity of the fluid are assumed to remain constant. Although some axial quantities are constant for all i, the notation is used to demonstrate how these quantities will change in the radial case, for each j. In the case of the solid, this advection term is not present. The thermal movement from advection is calculated in Equations 4.10 and 4.11:

$$\frac{dT}{dt}_{f,ad,i} = \frac{\rho_f u_{ax,i}}{m_{f,i} C_{p,f,i}} \left(A_{ax,i-1} T_{f,i-1} C_{p,f,i-1} - A_{ax,i} T_{f,i} C_{p,f,i} \right) \tag{4.10}$$

$$\frac{dT}{dt}_{s,ad,i} = 0 \tag{4.11}$$

For Equations 4.10 and 4.11, each at cell (i, j) unless otherwise specified, ρ_f is the density of the fluid, $u_{ax,i}$ is the axial velocity, m_f is the mass of fluid within the cell and $C_{p,f}$ is the heat capacity of the contents of the cell.

Heat transferred through the walls by convection is calculated by Equation 4.12. Each fluid is in contact with some amount of the solid wall, as determined by $A_{u,i}$, but has no direct contact with the other fluid. In the case of the solid, contact is between the solid and both of the fluids simultaneously. The wall heat transfer coefficient, h, is an overall average from experimental data. While this parameter will not be constant over the entire domain, it is assumed that the differences are small enough to be ignored.

$$\frac{dT}{dt}_{f,conv,i} = \frac{h_1 A_{p,i}}{m_{f,i} C_{p,f,i}} \left(T_{s,i} - T_{f,i} \right) \tag{4.12}$$

$$\frac{dT}{dt}_{s,conv,i} = \frac{h_1 A_{p,i}}{m_{s,i} C_{p,s,i}} \left(T_{f1,i} - T_{s,i} \right) - \frac{h_2 A_{p,i}}{m_{s,i} C_{p,s,i}} \left(T_{s,i} - T_{f2,i} \right) \tag{4.13}$$

Internal heat conduction within the fluid is calculated by the term $\frac{dT}{dt}_{f,cond,i}$. For systems with higher velocities and hence Reynolds number, this term will likely be negligible.

$$\begin{aligned} \frac{dT}{dt}_{f,cond,i} = \frac{1}{m_{f,i} C_{p,f,i} \Delta z} \big(& k_{f,i-1} A_{ax,i-1} T_{f,i-1} - 2 k_{f,i} A_{ax,i} T_{f,i} \\ & + k_{f,i+1} A_{ax,i+1} T_{f,i+1} \big) \end{aligned} \tag{4.14}$$

$$\begin{aligned} \frac{dT}{dt}_{s,cond,i} = \frac{1}{m_{s,i} C_{p,s,i} \Delta z} \big(& k_{s,i-1} A_{ax,i-1} T_{s,i-1} - 2 k_{s,i} A_{ax,i} T_{s,i} \\ & + k_{s,i+1} A_{ax,i+1} T_{s,i+1} \big) \end{aligned} \tag{4.15}$$

Analogous equations may be formed for the radial movements of heat throughout the tube, in which the term Δz must be substituted with Δr. These differ only in boundary conditions used. In cases where the flow is parallel with the main axis of the tube, the radial velocity will be zero, and the advective term may be ignored. The radial components of the thermal movements are added to those of the axial to give the net change for each of the thermal quantities.

$$\frac{dT_{f1}}{dt} = \frac{dT_{f1,i}}{dt} + \frac{dT_{f1,j}}{dt} \tag{4.16}$$

$$\frac{dT_{f2}}{dt} = \frac{dT_{f2,i}}{dt} + \frac{dT_{f2,j}}{dt} \tag{4.17}$$

$$\frac{dT_s}{dt} = \frac{dT_{s,i}}{dt} + \frac{dT_{s,j}}{dt} \tag{4.18}$$

Due to the limitations of the MATLAB model, boundary conditions are only imposed on the extremes of the domain in each of the axial and radial directions. The establishment of specific features, such as a shell with a non-uniform outer geometry was not possible while maintaining generality in the equations used. Boundary conditions in the axial direction consist of a Dirichlet boundary at the inlet, and a Neumann boundary at the outlet. In the case of co-current flow, these equations are identical for each of the fluids. Conversely, for counter-current operation, the boundary locations are swapped for one of the fluids.

$$\begin{aligned}\frac{dT_i}{dt}_{i=1} &= \frac{\rho_f u_{ax,i}}{m_{f,i} C_{p,f,i}} \left(A_{ax,in} T_{f,in} C_{p,f,in} - A_{ax,i} T_{f,i} C_{p,f,i}\right) \\ &+ \frac{h_1 A_{u,i}}{m_{f,i} C_{p,f,i}} \left(T_{s,i} - T_{f,i}\right) \\ &+ \frac{1}{m_{f,i} C_{p,f,i} \Delta r} \left(-k_{f,i} A_{ax,i} T_{f,i} + k_{f,i+1} A_{ax,i+1} T_{f,i+1}\right)\end{aligned} \tag{4.19}$$

$$\begin{aligned}\frac{dT_i}{dt}_{i=m} &= \frac{\rho_f u_{ax,i}}{m_{f,i} C_{p,f,i}} \left(A_{ax,i-1} T_{f,i-1} C_{p,f,i-1} - A_{ax,i} T_{f,i} C_{p,f,i}\right) \\ &+ \frac{h_1 A_{u,i}}{m_{f,i} C_{p,f,i}} \left(T_{s,i} - T_{f,i}\right) \\ &+ \frac{1}{m_{f,i} C_{p,f,i} \Delta r} \left(k_{f,i-1} A_{ax,i-1} T_{f,i-1} - 2 k_{f,i} A_{ax,i} T_{f,i}\right)\end{aligned} \tag{4.20}$$

Boundary conditions in the radial direction consist of Neumann boundaries, for both extremes of the cylindrical radius. The point $j = 0$ represents the central point to the fluid flow through the cylinder. As it is assumed that the thermal profile for the heat exchanger is radially

symmetrical, it is not possible for fluid to pass back over this centre line, as doing so would violate this symmetry. At the point $j = n + 1$, a well insulated, adiabatic outer wall exists.

$$\frac{dT_j}{dt}_{j=1} = \frac{\rho_f u_{r,j}}{m_{f,j} C_{p,f,j}} \left(-A_{r,j} T_{f,j} C_{p,f,j} + A_{r,j+1} T_{f,j+1} C_{p,f,j+1}\right) + \frac{h_1 A_{p,j}}{m_{f,j} C_{p,f,j}} \left(T_{s,j} - T_{f,j}\right) + \frac{1}{m_{f,j} C_{p,f,j} \Delta z} \left(-k_{f,j} A_{r,i} T_{f,j} + k_{f,j+1} A_{r,j+1} T_{f,j+1}\right) \tag{4.21}$$

$$\frac{dT_j}{dt}_{j=n} = \frac{\rho_f u_{r,j}}{m_{f,j} C_{p,f,j}} \left(A_{r,j-1} T_{f,j-1} C_{p,f,j-1} - A_{r,j} T_{f,j} C_{p,f,j}\right) + \frac{h_1 A_{p,j}}{m_{f,i} C_{p,f,j}} \left(T_{s,j} - T_{f,j}\right) + \frac{1}{m_{f,j} C_{p,f,j} \Delta z} \left(k_{f,j-1} A_{r,j-1} T_{f,j-1} - 2k_{f,j} A_{r,j} T_{f,j}\right) \tag{4.22}$$

The overall heat transfer coefficient U, between the heat exchanger channels is calculated according to experimental data (Symon, 2017) using the wall heat transfer coefficients, h and the thermal conductivity of the wall, k, for a given wall thickness Δx_w. This heat transfer determines the overall performance of the heat exchanger at transferring heat between the fluid streams.

$$\frac{1}{U} = \frac{1}{h_1} + \frac{\Delta x_w}{k} + \frac{1}{h_2} \tag{4.23}$$

It was found that the developed finite differencing method was incapable of adequately capturing the behaviour of any of the heat exchanger systems. This was due to the lack of shape effects, vital to the overall performance of any realistic heat exchanger. While not used, this initial work reinforced many of the concepts of fluid flow, heat flow and boundary conditions that need to be understood when any method is used. The MATLAB FDM approach was found to be unsuitable primarily due to computational considerations, whereby even small simulation domains would take many hours to process. In addition, the assignment of boundary conditions in specific locations and the implementation of physics on walls was hindered by the lack of an object-oriented programming approach. Parallel processing, while possible within MATLAB, is non-trivial to implement in situations where the separated memory cores must interact.

4.3 Lattice Boltzmann method

The LBM, as described in Chapter 2, was implemented through the use of the Palabos framework. Palabos is an open source set of libraries designed to aid in the writing of parallel C++ lattice Boltzmann codes. The Palabos framework provides both a large assortment of multi-physics behaviours that may be actively altered by the user, as well a large amount of backend programming designed to optimise codes for massively-parallel applications through the use of the Message Passing Interface (MPI). Regardless, a large amount of C++ code was still required to be written on the frontend of code development, to accurately tie together various dynamics, and implement new ones. The base Palabos libraries utilised were similar to those used by Dolamore (2017) and Houlton (2019).

A lattice of uniform nodes across three dimensional space. In the present work, each node may be accessed from 18 directions, each of which have a population density tracked. The central node also has a density tracked, making 19 components for each lattice node, hence giving the name D_3Q_{19}. This is illustrated in Figure 4.2.

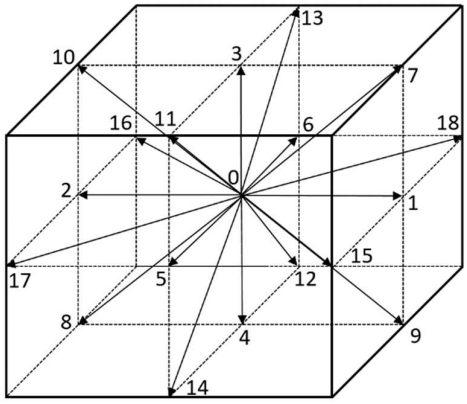


Figure 4.2. Orientation of the D_3Q_{19} lattice.

The collision equation is the core of the LBM, as the main driving equation advancing the system. In the present work, the double distribution function method is used, whereby one lattice, f, tracks the density of the fluid, while a second advection diffusion lattice, g, tracks the energy density, expressed in macroscopic form as temperature. Equations 4.24 and 4.25 are the collision equations for the momentum and thermal lattices. The unit vector e_i is the discrete velocity in each lattice direction; $\delta_x = \delta_t = 1$ are the lattice space and time steps, in lattice units; τ is the relaxation period controlling the size of the time steps; f^{eq} and g^{eq} are equilibrium distribution functions, towards which the system will relax; F is a body force term

acting on the fluid, in this case the Boussinesq approximation; and Q is a heating source term, which can be added optionally (Mohamad, 2019).

$$f_i(\boldsymbol{x}+\boldsymbol{e_i}\delta_t, t+\delta_t) - f_i(\boldsymbol{x},t) = -\frac{1}{\tau_v}\left[f_i(\boldsymbol{x},t) - f_i^{eq}(\boldsymbol{x},t)\right] + \delta_t F_i(\boldsymbol{x},t) \tag{4.24}$$

$$\begin{aligned} g_i(\boldsymbol{x}+\boldsymbol{e_i}\delta_t, t+\delta_t) - g_i(\boldsymbol{x},t) \\ = -\frac{1}{\tau_g}\left[g_i(\boldsymbol{x},t) - g_i^{eq}(\boldsymbol{x},t)\right] + \left(1-\frac{1}{2\tau_g}\right)Q_i(\boldsymbol{x},t) \end{aligned} \tag{4.25}$$

Each direction of the lattice has its own weighting, depending on spatial orientation. These weightings are unique to each number of lattice directions, but are otherwise independent of the system being studied. The weightings, w for the D_3Q_{19} lattice are given by:

$$w_i = \begin{cases} \frac{1}{3} & i = 0 \\ \frac{1}{18} & i = 1-6 \\ \frac{1}{36} & i = 7-18 \end{cases} \tag{4.26}$$

The discrete velocity $\boldsymbol{e_i}$ for each direction of the D_3Q_{19} lattice are as follows in Equation 4.27:

$$\boldsymbol{e_i} = \begin{cases} (0,0,0) & i = 0 \\ (\pm1,0,0),(0,\pm1,0),(0,0,\pm1) & i = 1-6 \\ (\pm1,\pm1,0),(\pm1,0,\pm1),(0,\pm1,\pm1) & i = 7-18 \end{cases} \tag{4.27}$$

For the D_3Q_{19} lattice, the equilibrium distribution functions are given by Equations 4.28 and 4.29:

$$f_i^{eq} = w_i\rho\left[1 + \frac{\boldsymbol{e_i}.\boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{e_i}.\boldsymbol{u})^2}{2c_s^4} - \frac{\boldsymbol{u}^2}{2c_s^2}\right] \tag{4.28}$$

$$g_i^{eq} = w_i\theta\left[1 + \frac{\boldsymbol{e_i}.\boldsymbol{u}}{c_s^2}\right] \tag{4.29}$$

where ρ is the physical density of the fluid, u is the physical velocity, θ is the dimensionless temperature and $c_s = 1/\sqrt{3}$ is the speed of sound in lattice units for the fluid. The quantities ρ and θ are macroscopic properties which may be calculated as the zeroth order moment of each of their respective lattices, as in Equations 4.30 and 4.31.

$$\rho = \sum_i f_i \tag{4.30}$$

$$\theta = \sum_i g_i \tag{4.31}$$

From the lattice fluid density, the lattice gauge pressure may be determined from Equation 4.32:

$$P_{LB} = \rho c_s^2 \left(P_{amb,LB} - \rho_{LB} c_s^2 \right) \tag{4.32}$$

where the quantity $P_{amb,LB}$ is the gauge pressure in lattice units, and $\rho_{LB} = 1$ is the lattice reference density. The physical temperature T may be converted into the dimensionless temperature θ through the conversion:

$$\theta = \frac{T - T_C}{T_H - T_C} \tag{4.33}$$

where T_H and T_C are the hot and cold inlet temperatures respectively. These quantities also represent the physical limits of the temperature range for the simulations, limiting the range of possible values of the dimensionless temperature to the domain [0,1]. This is the domain for which numerical stability will be highest, as well as preventing potential issues arising due to fluids passing below 0°C.

Equations 4.30 and 4.31 state that the summation of the populations in each of the Q possible directions at each node will give the population density. Although LBM is primarily used within fluids, the solid portions of the domain are still tracked in the same manner. Within the solid, the equilibrium distribution function differs only in that the velocity terms are set to zero. With all velocity components cancelled, Equation 4.34 is greatly simplified:

$$g_{si}^{eq} = w_i \theta \tag{4.34}$$

The relaxation time for the hydrodynamic lattice is set by default to a value of $\tau_v = 1$ in dimensionless units. Although some deviation from this is allowable, the system is generally the most stable with this parameter near unity (Zhao, 2013). Zhao discusses the effects of altering the relaxation time to find an optimum between computational speed, stability and numerical accuracy for a single-fluid simulation. They found that the stability and efficiency both increase as the relaxation time is lowered. The remaining dimensionless groupings, such as the Mach number must be considered however, to ensure that a balance is struck to maintain accuracy. Simulations will become unstable when the value of the relaxation time is outside of the range $0.55 < \tau < 2.0$, with an approximate optimum of $\tau = 0.81$ for their application. A

relaxation time of unity was used for the present work for simplicity. The kinematic viscosity ν_f, may be determined from Equation 4.35:

$$\nu_f = (\tau_v - 0.5)c_s^2\delta_t \tag{4.35}$$

To provide a link between the hydrodynamic lattice and the thermal lattice, the Prandtl number, Pr is used. The Prandtl number represents the ratio of momentum diffusivity to the diffusivity of heat within a fluid and is represented by Equation 4.36:

$$Pr = \frac{\nu_f}{\alpha_f} \tag{4.36}$$

Since the Prandtl number is fixed for the fluid in use, the thermal diffusivity of the fluid may be determined. The relaxation period of the thermal lattice, τ_g, may be calculated by Equation 4.37:

$$\alpha_f = \left(\tau_g - 0.5\right)c_s^2\delta_t \tag{4.37}$$

A similar method must be employed to find the thermal diffusivity of the solid. Since the thermal diffusivity of the fluid is known, and ratios are maintained regardless of dimensions, a ratio of the thermal properties of the solid to those of the fluid is created. The thermal diffusivity, α is made up of the quotient of the thermal conductivity, k, and the heat capacitance term ρc_p.

$$\alpha = \frac{k}{\rho c_p} \tag{4.38}$$

The final relaxation period required for the system, τ_{gs}, may then be found from Equation 4.39:

$$\alpha_s = \left(\tau_{gs} - 0.5\right)c_s^2\delta_t \tag{4.39}$$

4.3.1 Dimensional Scaling

To improve the numerical stability of the LBM simulations, calculations are carried out with dimensionless lattice units, as denoted by the subscript LB. All other quantities use physical units. The dimensionless quantities are only converted back to physical quantities for visualisation of the results. Each of the aforementioned lattices share a spatial resolution N, while maintaining different time scales. For the forced convection problem being simulated, the Reynolds number Re is used to find the relaxation period for the hydrodynamic lattice.

$$Re = \frac{L_{char}|u_{in}|}{\nu_f} \tag{4.40}$$

The characteristic length L_{char} has a number of definitions, but the version used in Equation 4.40 is the physical length of a single TPMS unit cell. The conversion between unit systems is achieved through the use of dimensionless numbers. While most quantities will change in value based on the dimensional scaling used, a small number of dimensionless quantities and ratios are conserved. These dimensionless quantities include the Reynolds number, Prandtl number, geometric aspect ratio and the thermal diffusivity ratio. As such the Reynolds number may also be formulated based on dimensionless quantities:

$$Re = \frac{N|u_{LB}|}{\nu_{f,LB}} \tag{4.41}$$

An alternative form of the physical formulation for the Reynolds number utilises the hydraulic diameter D_H as the characteristic length of a unit cell. Within a classical pipe, the hydraulic diameter is defined as:

$$D_{H,pipe} = \frac{4A_c}{L_p} \tag{4.42}$$

where A_c is the cross-sectional area of the pipe, and L_p is the wetted perimeter of the same cross-section. The cross-sectional area is not constant for all TPMS structures however, encouraging the use of a form averaged across the entire structure. As described by Shamey and Zhao (2014), another possible definition for the hydraulic diameter used in the study of porous media, is as follows:

$$D_{H,pore} = \frac{4V_f}{A_S} = \frac{4\epsilon}{SSA} \tag{4.43}$$

The Reynolds number in Equation 4.43 is applicable for a wider range of porous structures, such as those where the channel width is non-uniform. While the most rigorous, the surface area exposed to the fluid can be difficult to calculate. Sometimes as a very simple alternative, the average channel width may be used. The version from Equation 4.44 was used in this work.

$$L_{char} = D_{H,laminar} = D_{ave} \tag{4.44}$$

Assuming the dimensional conversions are carried out correctly, the value of the Reynolds numbers calculated by Equations 4.40 and 4.41 should be equal. The unit of length dx is the length of each of the sides for the simulation unit cell, of which are all equal. This unit length

is affected only by the resolution of the simulation chosen. In general, higher resolutions provide greater accuracy, but come at greater computational cost. In lattice units this quantity is expressed as:

$$dx = \frac{L_{char}}{N} \tag{4.45}$$

where N is the number of computational nodes in a single direction. The time scale dt is linked to the Reynolds number, the resolution and the kinematic viscosity. Equation 4.46 gives the formula for which this scale is determined for the hydrodynamic lattice:

$$dt = \frac{\nu_f Re}{N^2} \tag{4.46}$$

The lattice units used for simulation may be simply converted back into physical units by reversing the process by which they were formulated. Equations 4.47-4.49 give these conversion:

$$u = u_{LB} \frac{dt}{dx} \tag{4.47}$$

$$P = P_{LB} \rho_{phys} \frac{dx^2}{dt^2} \tag{4.48}$$

$$T = \theta(T_H - T_C) + T_C \tag{4.49}$$

Ideally, to ensure stability, the relaxation time parameters τ_v and τ_g should each have values of approximately unity. While the hydrodynamic lattice can have parameters chosen such that this is the case, the process cannot be simultaneously done for the thermal lattice, due to the diffusivity links between the two. Although altering the relaxation time may have detrimental effects on the stability of the hydrodynamic lattice, in appropriate circumstances the performance of the thermal lattice may be improved. Care must still be taken to not reduce the value of the relaxation period too much, as a value of $\tau_v < 0.55$ will tend to be unstable (Zhao, 2013). For simulations with resolutions within a feasible range, of approximately $N \approx 120$, the Reynolds number must be restricted to prevent the lattice velocity from approaching a significant fraction of the speed of sound in the lattice. For this reason, the Reynolds numbers investigated within this work are restricted to values in the region of $Re \approx 1$.

As used in Equation 4.47, for a given spatial resolution and Reynolds number, the values of the lattice kinematic viscosity ν_{LB} and the lattice velocity u_{LB} are each free variables. While the kinematic viscosity influences only the relaxation period, the lattice velocity provides

another tuning mechanism, albeit also with limits. The maximum lattice velocity that may be simulated for a given resolution is limited by the speed of sound within the lattice, c_s. The speed of sound provides an upper limit on the stable rate of information transfer across the system, and compressibility errors exist for velocities approaching it. While methods do exist to partially bypass these limits and model compressible flows by the LBM (Alexander *et al.*, 1992), they are not implemented for complexity and stability reasons within this work. Provided that the condition $|u|_{LB} < 0.1c_s$ is followed, the velocity is low enough that the flow may be treated as incompressible. This provides an upper limit on the value of the lattice velocities that may be chosen, despite being a free parameter.

In order to further dynamically couple together the hydrodynamic and thermal lattices, the Boussinesq correction could be applied. This correction imposes a buoyancy force on the fluid, according to the temperature induced density changes occurring. Usually, the Boussinesq approximation is applied by way of a force correction to the F term in Equation 4.24. However, Imani (2017) showed that making an adjustment to the velocity used during the collision step was actually more computationally efficient and accurate than a force correction alone. As such, Equation 4.50 was used to account for the variations in density caused by the spatially changing temperature of the fluid, where β is the coefficient of thermal expansion and g_{acc} is gravitational acceleration.

$$\boldsymbol{u} = \frac{1}{\rho}\left(\sum_{i=0}^{\mathrm{Q}-1} f_i \boldsymbol{e_i} + \rho g_{acc}\beta\left(T_F - T_{Ref}\right)\tau_v \delta_t\right) \tag{4.50}$$

Equation 4.50 was the version implemented at the collision step only. Elsewhere, the simplified version in Equation 4.51 was used instead. This quantity is the first order moment of the fluid lattice.

$$\boldsymbol{u} = \frac{1}{\rho}\sum_{i} f_i \boldsymbol{e_i} \tag{4.51}$$

4.3.2 Boundary Conditions

The location of the boundaries for each of the domain entrances, two inlets and two outlets, were identified in a partly automated manner according to the STL file input. On each entrance, one vertex point must be chosen at approximately the centre of each entrance tube. This vertex is used as the centrum for a circular cap, constructed from triangles similar to an STL file,

placed over the entrance in question. Any cell on the domain edge coincident with a projection of these capping triangles from their normal, is designated as one of the entrance boundaries. The order in which these boundaries will be identified is supplied by the user, allowing inlets and outlets to be set.

The geometries generated in Chapter 3 represent the domains on which the simulations were carried out. These geometries come in the form of either a full-scale domain including the entire heat exchanger structure, or a limited unit cell structure. Chapter 3 detailed a voxelisation algorithm which was used to convert the TPMS equations first into a voxel field, and then into an STL structure consisting of many interconnected triangles. Within the present C++ based program, the process is reversed, and the triangular structures are reformed into voxel fields. The process of converting into, and then back out of the STL file format is computationally inefficient compared with generating the simulation domain directly from the applicable TPMS equations. However, maintaining the functionality to simulate directly based on STL structures greatly enhances the usefulness of the program. Simulation domains may be taken from the same STL files as those being sent to 3D printers, removing sources of error when attempting to recreate the part. Another advantage to utilising the STL file format is the ability to augment structures in SOLIDWORKS, with components not easily defined by geometric equations.

The purpose of the voxelization process is to identify which parts of the geometry are intended to be solid, and the parts that should be fluid. By seeding the algorithm with the fact that the outermost cells are always fluids, the first walls from the STL file encountered always have solids on the opposite side. Guaranteeing that a fluid is always encountered first prevents the case where a geometry could be formed with the inverse of the expected fluid-solid distribution. In addition to the fluid and solid cells, border variants of each are placed in between. Although these border cells usually acted the same as their respective bulk domain counterparts, for the purposes of some boundaries they could be assigned different dynamics, or be used for locating features according to voxel type.

This same algorithm was used regardless of whether the simulation domain was a full-scale structure, or simply a unit cell. In the case of the full-scale model, it was desirable for the volume outside of the heat exchanger shell to be a fluid, to which heat could optionally be conducted through an imperfectly insulated outer wall. The outer domain being filled with fluids is acceptable for the full scale version, with the relatively small inlets and outlets being set as boundaries through their proximity to the pre-specified entrance points. For the unit cell case, this voxel seeding behaviour was problematic, as it was not desirable to have fluids

surrounding the entire geometry. The simplest solution to this problem was to remove the two outermost layers of cells from the domain, and move the boundaries to the new outermost cells. Doing so removes the superfluous fluid and border fluid layers, leaving the new outermost cells able to be differentiated from each other by the aforementioned algorithms.

On the interface between fluids and solids, bounce back nodes are placed, as described by Mohamad (2019). On these nodes, population density approaching from any direction is reflected back in the direction it originated from, as described in Equation 4.52, and graphically in Figure 4.3. The quantity $\hat{f}_i$ denotes the population in a single direction after collision.

$$\hat{f}_i = f_{i+\frac{Q-1}{2}} \tag{4.52}$$

For a bounce-back node on the surface, the fluid populations moving in the direction of the solid are known from the streaming at the previous time step. Those populations moving in the opposite direction are then set to be equal, as according to Equation 4.52. By reversing the direction of the populations originating from the next closest fluid layer of cells, the populations are effectively cancelled out. This has the effect of setting the macroscopic velocity of this layer to zero. The zero velocity on the surface matches the no-slip behaviour expected of the surface at low Reynolds number.

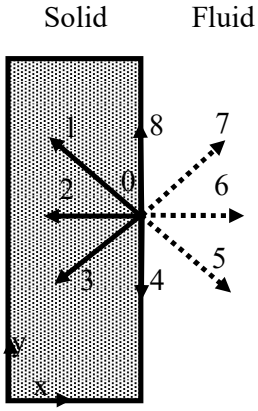


Figure 4.3. Effects of the bounce back nodes.

An adiabatic boundary was created along the outer surface of the heat exchanger, to mimic the effects of a layer of insulative material on the exterior. It was assumed that this layer would be additional to the original structure supplied in the STL file, and would be formed of a material with very low thermal conductivity. The boundary was realised as follows: during initial setup of the simulation, an extra scalar field containing a singular integer for each lattice point, populated with the codified information of the cell types of all cells surrounding the heat

exchanger exterior, while leaving the interior blank. Inside a parallelised data processor, the domain was first searched for solid border cells, then the neighbours of each of those cells testing positive were checked for being fluid cells, all within the scalar field. When the aforementioned cells were confirmed to be fluid cells, their populations moving in that direction were recorded and set to zero. Those moving in the opposite direction back away from the boundary had the mean value of those pre-set populations added to them. This process occurred independently on every node of the heat exchanger exterior, as denoted by Equations 4.53 and 4.54 (Mohamad, 2019):

$$\hat{g}_{i+\frac{Q-1}{2},new} = \hat{g}_i + \hat{g}_{i+\frac{Q-1}{2},old} \tag{4.53}$$

$$\hat{g}_i = 0 \tag{4.54}$$

where the direction i is that of a fluid cell neighbouring a border solid. Boundary corrections must be applied after the collision step, due to the changes made at this point, and before the streaming step of the LBM algorithm, when the alterations made are propagated through adjacent cells.

The inlets are Dirichlet boundaries whereby on the applicable edges, the unknown outgoing populations are set equal to the incoming populations from the previous time step, plus momentum weighted according to the directional weightings of the lattice. These form a system of simultaneous equations, from which each of the outgoing populations may be solved. An example of the form of the boundary used is given in Equations 4.55-4.57, for the much simpler D_2Q_9 lattice, where an inlet is positioned facing the positive x direction, using the numbering scheme from Figure 4.3. In this case, the populations f_5, f_6 and f_7 are the unknowns to be found (Mohamad, 2019).

$$f_5 = f_5^{eq} + f_1^{eq} - f_1 \tag{4.55}$$

$$f_6 = f_6^{eq} + f_2^{eq} - f_2 \tag{4.56}$$

$$f_7 = f_7^{eq} + f_3^{eq} - f_3 \tag{4.57}$$

Although the individual populations being calculated vary with the position of the boundary, the pattern will always remain the same. This applies as well to the D_3Q_{19} lattice, where more populations must be computed, but using the same rules.

For the outlets, many options exist for the implementation of a Neumann boundary, ranging from simple spatial conditions, to more complicated temporal boundaries. The chosen method,

one made available by Palabos, utilises a partial temporal boundary, including terms from the equilibrium portion of the current cell, as well as the sum of the populations from the previous time step. For each time step, the previous values of density and momentum are kept recorded, such that at the boundaries the density may be maintained constant.

$$f_{i,t} = f_{i,t}^{eq} + f_{i+\frac{Q-1}{2},t}^{eq} - \sum_{i} f_{i,t-1} \tag{4.58}$$

One of the advantages of this method is that the outlet is independent of the geometry. If a curved wall exists next to the boundary and one of the cells adjacent to the entrance becomes blocked, the outlet will still function as expected. The downside comes computationally, where a copy of the hydrodynamic lattice is made, and the zeroth order moment computed at each time step. Other more computationally expensive algorithms exist, but these were not chosen due to stability for the chosen geometry.

Appropriate initial conditions were important for the each of the entrances to function correctly. These varied for the thermal lattice depending on whether the simulation was dynamic or at steady state. For dynamic operation, all cells were set to initially begin at the minimum dimensionless temperature of $\theta = 0$. Heat ingress originates from the hot channel, and is mitigated by the movement of cold fluids through the cold channel. For steady state operation, the fluid temperatures were instead initialised to the temperatures at which their respective inlet channels were operating. Doing so increased stability and reduced the time required to attain the steady solution. For the hydrodynamic lattice, the domain was split into two separate regions. Those cells deemed to be within the structure, as determined by the scalar field created earlier were initialised with the velocity at the inlet. Those outside were left with zero velocity. This was done as these nodes are not required for the simulation, and to help prevent purely numerical artefacts.

Parallelisation was handled within Palabos by restricting each of the computational cores of the machine used to certain bounds within the simulation domain, balanced such that they were approximately equal. Envelopes of overlap exist between the domains of each thread, allowing for single neighbour searches to be carried out throughout the domain. A limitation was applied that any algorithmic search for a cell greater than one unit distance away could only be found under the condition that a non-parallelised field was being searched. Within a parallelised field, cells outside of this range would be inaccessible to the CPU process carrying out the computations in that region of the domain. Such non-parallel fields required to be accessed by

all processes were kept to a minimum, due to the much high computational cost of creating them.

4.3.3 Conjugate Heat Transfer

A conjugate boundary is one where both a fluid and a solid are present, but no information is known *a priori*, as in the case of Dirichlet or Neumann boundaries. Instead, the continuity condition must be applied:

$$T^+ = T^- \tag{4.59}$$

$$\boldsymbol{n}\left[k\nabla T + \rho c_p u T\right]^+ = \boldsymbol{n}\left[k\nabla T + \rho c_p u T\right]^- \tag{4.60}$$

Equations 4.59 and 4.60 state that the temperature and heat flux, consisting of both conductive and convective components, on each side of the boundary must be equal, where n is a unit vector normal to the boundary. The conjugate boundary implemented in the current work is that used by Lu *et al.* (2017), adapted for 3D, similar to the enactment from Imani (Imani *et al.*, 2012; Imani, 2017). Here, the concept of a counter-slip energy (D'Orazio & Succi, 2003) was applied to find the missing populations on the fluid-solid boundary. The present work makes use of a layer of boundary cells between the solid and fluid thermal lattices, upon which the boundary itself was situated, as outlined earlier.

The actual implementation of the conjugate boundary, as it pertains to LBM is as follows: first, the values of the thermal lattice are recorded prior to the collision step, so that a record exists after the originals are gone. The domain is then split into bulk fluids, bulk solids, and those cells on the boundary. After the collision and streaming steps, the thermal lattice is corrected such that the solid portion of the thermal lattice takes in a different quantity of heat to what was passed to it from the fluid. This change accounts for the differing heat capacitance (mc_p) between the two materials. The first step, performed on every cell along the boundary, is the calculation of the boundary temperature from the following equation:

$$T_b = \frac{k_a T_a + k_c T_c}{k_a + k_c} \tag{4.61}$$

Here, the point a is a fluid node along the boundary, c are solid nodes, while b are purely theoretical points that lie between them. With the temperatures known, the equilibrium distributions at each of the points could be calculated in the usual manner, similar to Equation 4.29, which may be used for point c for the term $g_{c,i}^{Eq}$. For the boundary cells the no slip

condition is used, making all velocity terms equal to zero. As a result, Equation 4.61 may be used instead to find g_b^{Eq}. The next step is to approximate the non-equilibrium portion at each of the points b. The total particle population is approximated as the linear sum of its equilibrium and non-equilibrium portions (Zhao-Li *et al.*, 2002). The non-equilibrium portion of the population on the boundary, in the direction away from a fluid cell, is approximated as being the same as that originating from the aforementioned fluid cell. This is represented by Equation 4.62:

$$g_{b,i}^{Neq} = g_{c,i} - g_{c,i}^{Eq} \tag{4.62}$$

Collision of the boundary cells is then carried out, through a version of the ordinary collision equation:

$$\hat{g}_b = g_b^{Eq} + \left(1 - \frac{1}{\tau_v}\right) g_b^{Neq} \tag{4.63}$$

The final values for the correction to the fluid cells, post streaming, were as follows:

$$g_{c,i} = 2\hat{g}_b - \hat{g}_{c,i+\frac{Q-1}{2}} \tag{4.64}$$

This scheme is implemented only on cells internal to the heat exchanger structure; the shell is excluded, and instead subject to the adiabatic boundary. A Palabos data processor function was written to make the aforementioned alterations to the thermal lattice population distribution function as given by Equation 4.25 after each time step was evaluated.

4.3.4 Representative elemental volume

An alternative, REV approach was investigated, with the goals of reducing computation times and increasing simulation stability. By reducing the scope of the simulation from the entire heat exchanger structure down to just a small number of unit cells, higher resolutions per unit cell were possible. These higher resolutions allowed for greater numerical stability, as well as the testing of different dynamics that did not work well in the previous model. The largest change made was to the domain being studied. In the REV version, the shell was ignored, and a sheet TPMS unit cell studied instead. This new domain shown in Figure 4.4 was assumed to be representative of any other within the structure.

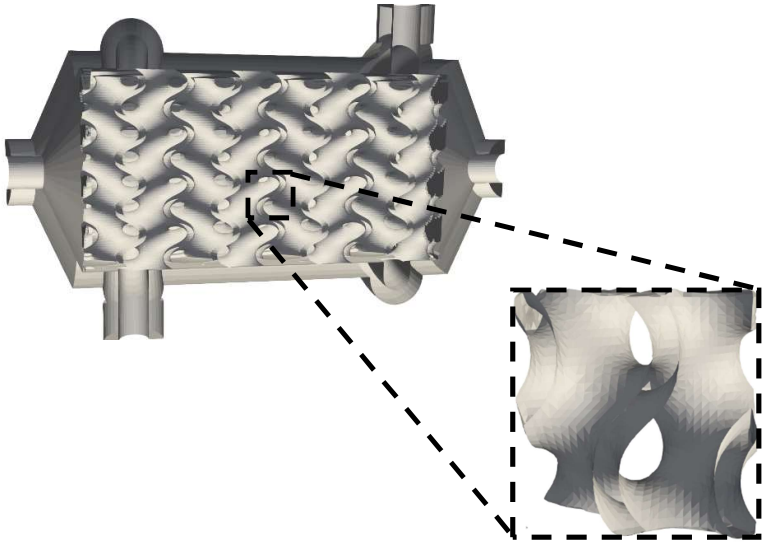


Figure 4.4. The domain included within the REV approach.

While greater stability and speed of computation were gained by the REV approach, the ability to check the effects of some changes to parameters or greater structures was lost. By this method, it was no longer possible to easily examine the effects of changing the properties of the shell. Examples of these include the geometry of the shell and the packing within it, heat losses from non-perfect exterior insulation, or the placement of the heat exchanger entrances.

The boundary conditions for the REV approach were assigned differently to those of the full domain model. The REV version has no obvious inlet or outlet pipes as the full domain model does. Instead, the domain is voxelised, as earlier, and a scalar field is constructed. Initially the scalar field is filled with the wall locations only and the fluid cells remain undetermined. The voxelisation algorithm required that the domain be placed within a margin of fluid cells. Any object within this domain, would then also form an additional layer of border fluid cells between the fluid and any solids. As described earlier, the placement of these margin cells meant that a fluid gap was always present around the exterior of any shapes. For this reason, the outer two layers of the domain were removed, separating the channels. Some care regarding the particular geometry was required at this step, as a TPMS structure with walls too thick, could be voxelised incorrectly at lower resolutions. This would result in dead zones within regions of the TPMS structure, and the possible failure to identify all cells within a particular channel as those belonging to the correct fluid.

After specifying that the midpoint on the first axial slice of the domain, or the closest fluid neighbour to it, was a fluid from the first channel system, a neighbour search algorithm was employed to fill out all contiguous, previously undetermined cells within the scalar field as that

same fluid. After a small number of repetitions to ensure a complete assignment, all remaining cells in the scalar field were set to be of the second channel system. With the complete contents of both channel systems identified across the entire domain, inlet and outlet boundaries could be easily identified on the domain edges in the z direction. All other edges were assigned to be periodic boundaries, as it was assumed that this single unit of simulation was representative of any other.

As previously described, voxelisation dead zones could form, due to the regions of the structure being inaccessible directly. It is notable that the these voxelisation dead zones are not inaccessible to the fluids of the simulation, due to the presence of periodic boundaries allowing access. A method was devised to check all unallocated cells on the edges of the domain after the first round of channel allocation. Any neighbours to such a cell on the opposite side of the domain, reachable through a periodic boundary, are also allocated to that same fluid. At that point, the allocation algorithm is continued, to reach any remaining cells within the previously dead region.

Although the TPMS structures are designed to have equal volumes for each channel system, the cross-sectional area for each channel within any given planal slice is not guaranteed to be equal. Figure 4.5 demonstrates a unit cell of the Schwarz primitive TPMS structure. The areas exposed on the slice of the xy plane at the axial position of $z = 0$ differ. One channel has access to only the small area on the interior of the geometry, while the second fluid takes up the entire outer domain.

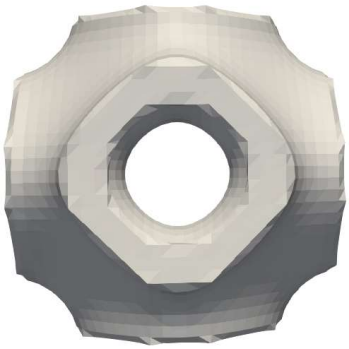


Figure 4.5. Cross section of a Schwarz Primitive TPMS structure at the position $z = 0$.

Due to the velocity inlet boundary being present on the $z = 0$ slice of the simulation domain, allocating the same velocity to all cells on the boundary would result in varying flow rates within each channel in cases where the cross-sectional areas of each differ. Since the overall channel volume is designed to be 50% for each, other slices of the unit cell will result in the opposite result as seen in Figure 4.5, whereby the area of the second channel will be larger.

Within these slices, the velocities will no longer be equal, as the cross-sectional area has been increased, meaning that neither velocity nor flow rate was conserved. For this reason, the velocity of the inlet cells is scaled when implementing the inlet boundary condition, as according to the area of cells available for each channel, after counting the total fluid and total solid cells on the inlet boundary side. Doing so ensures an equal flow rate passing into each channel, and an equal velocity of fluid averaged over the entirety of the structure.

When the differences in thermal diffusivities for each material are high, additional stability measures may be required. For this reason, the dynamics used for the thermal lattice were changed to those used by Perko and Patel (2014). The major difference between these dynamics and those used previously is the transfer of part of the thermal diffusivity into a velocity term u_D. Doing so reduces the effective discrepancy in relaxation parameters, improving those cases where the difference is sizable. These two velocities, from the hydrodynamic field, u_a and that from the diffusivity, u_D are combined as a simple sum wherever the velocity is required. Equations 4.65 and 4.66 give the calculation and usage of these terms:

$$u_D = \frac{\frac{\tilde{\alpha}}{\tau_g c_s^2}}{\left(1 + \frac{\tilde{\alpha}}{\tau_g c_s^2}\right)} \left(\frac{\sum_i g_i e_i}{\theta} - u_a \right) \tag{4.65}$$

$$u = u_D + u_a \tag{4.66}$$

where the thermal diffusivity term $\tilde{\alpha}$ is the non-reference portion of the total thermal diffusivity. These terms may be recombined by Equation 4.67, again as a simple sum in the same manner as Equation 4.66. The changes made by the algorithm depend entirely on the value chosen for the reference diffusion term α_{ref}.

$$\alpha = \alpha_{ref} + \tilde{\alpha} \tag{4.67}$$

Perko & Patel found that while physically, the choice of the reference diffusivity was meaningless, numerically, gains could be made to simulation stability by choosing higher values for the parameter. Therefore, by default the highest thermal diffusivity material is usually chosen as the value for the reference diffusivity. This value may be increased, potentially by many orders of magnitude for higher numerical accuracy, while reductions to the value will increase the speed of computations. In this work, the reference diffusivity value is always set to that of the material with the highest thermal diffusivity, usually the solid phase.

5 Verification

5.1 Overview

A number of numerical verification simulation examples were created to test the accuracy of the LBM model employed against known results from other studies. Testing is important to ensure that the solution method employed does not deviate greatly from other methods. Numerical testing also includes the establishment of the parameter ranges that may tested, and the effects that any changes to the simulation resolution may incur. For most examples, the properties of the system at steady state are of most interest. The performance of the numerical methods on an unstable system is also tested.

The main numerical test case is a hollow cube, designed to replicate baseline tests used by Imani (2017), Frederick and Moraga (2007) and others. One wall of the cube, positioned at $x = 0$ is hot, while the opposite wall at $x = H$ is cold, and the remaining walls are adiabatic. The box has all sides of length H, upon which the fin is positioned at the centre of the yz plane. The fin is of width $h = H/2$ in the y direction, length $l = H/2$ in the x and thickness $\Delta x_w = H/10$ in z. These ratios are preserved through the process of non-dimensionalising the system and subsequent discretising back into lattice units. The air within the box moves by natural convection only. Gravity is oriented in the downward, negative z axis. There is a fin on one wall, as shown in Figure 5.1.

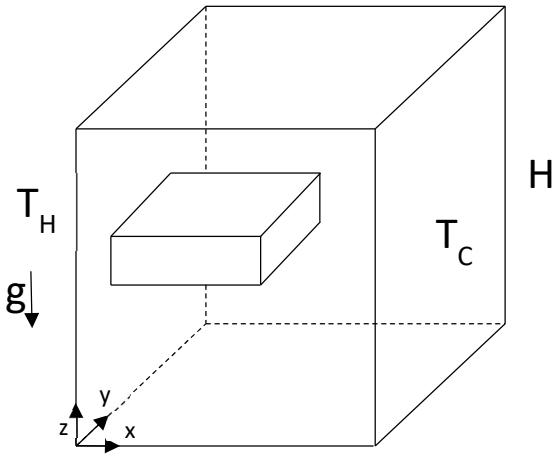


Figure 5.1. Finned enclosure benchmark.

As a natural convection case, Rayleigh number, given by Equation 5.1 is set instead of Reynolds number.

$$Ra = \frac{\rho \beta \Delta T H^3 g}{\nu \alpha} \tag{5.1}$$

Simulations were carried out with $Pr = 0.71$, $Ra = 10^5$ and $\lambda = 10$. The finned box was created in SOLIDWORKS to the specified dimensional ratios, before being exported as an STL file and voxelised for simulation. All simulations were run at a resolution of $N = 120$ cells in each direction in order to match the other studies being compared. Images were ccreated along the dimensionless lengths x^*, y^* and z^*. For the cubic enclosure, these were defined as follows:

$$x^* = \frac{x}{H} \tag{5.2}$$

$$y^* = \frac{y}{H} \tag{5.3}$$

$$z^* = \frac{z}{H} \tag{5.4}$$

The expected behaviour for a closed, natural convection system is for the vertical velocity to increase from zero against the hot wall, before rotating clockwise, being cooled by the cold wall and sinking again. The introduction of the fin both introduces a medium upon which heat may extrude further into the cube, as well as deflecting the convection current previously formed. The speed and degree to which heat may travel along the fin depend on the thermal properties assigned to it. For the steady state system, only the thermal conductivity ratio is important. Figure 5.2 show the steady state temperatures profiles for $Ra = 10^4$.

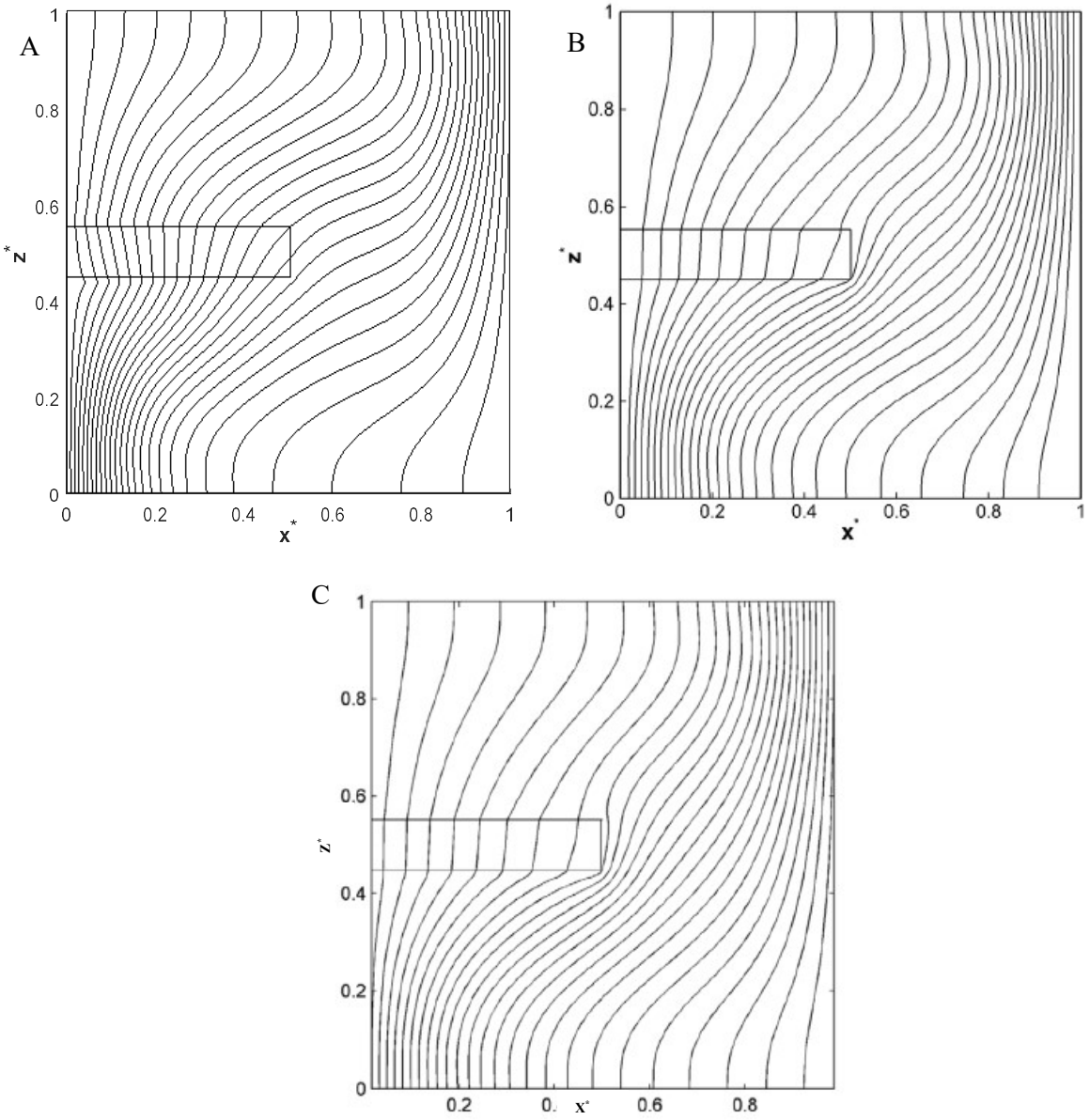


Figure 5.2. Temperature isotherms at steady state for $Ra = 10^4$ by A) the present work and B) Reproduced from Imani (2017) with permission from Elsevier, and C) Reproduced from (Frederick & Moraga, 2007) with permission from Elsevier.

At the lower Rayleigh number used in Figure 5.2, heat is transferred more slowly along the length of the fin, but the resulting temperature profiles of the fluids become very similar. This effect can be seen most noticeably at the tip. Similarly, the isotherms are also generated for $Ra = 10^5$ for the same finned enclosure in Figure 5.3:

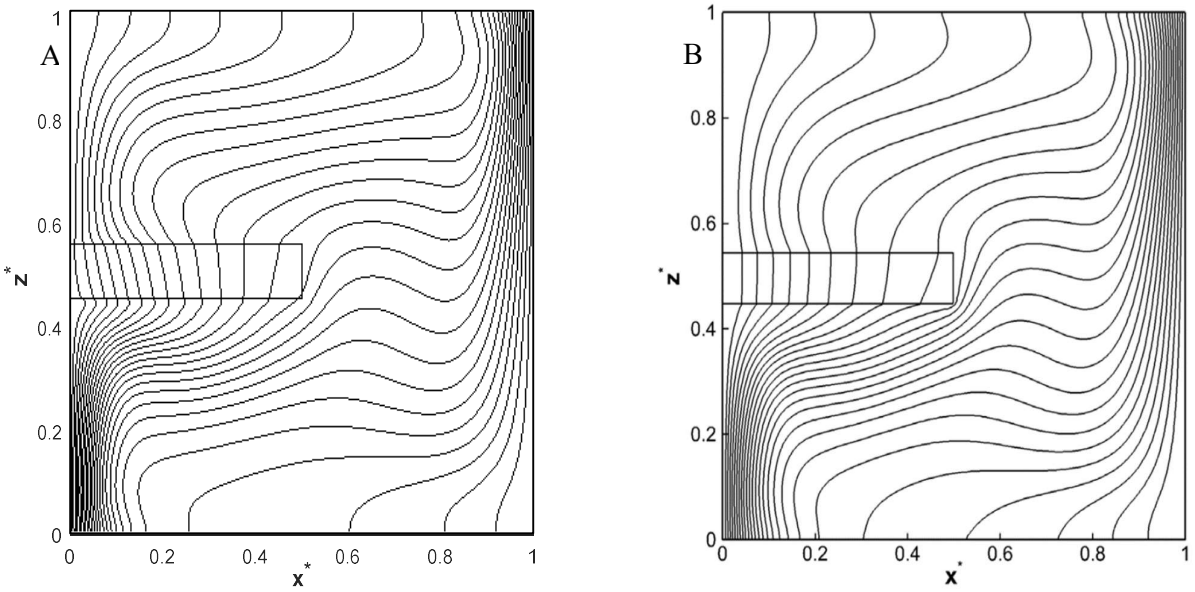


Figure 5.3. Temperature isotherms at steady state for $Ra = 10^5$ by A) the present work and B) Reproduced from Imani (2017) with permission from Elsevier.

Although good agreement is present in Figure 5.3 between the current work and that of the earlier study, minor differences exist. Compared with that of Imani in Figure 5.3, the current work appears to have a slightly higher velocity under the fin, warping the resulting temperature profile in the fluid slightly. Regardless, the overall temperature distribution of the present work very closely matches the benchmark.

5.2 Dynamics

Dynamics show greater differences due to the relatively fast changes that occur in the system at the start of the simulation. With the initial condition of $T = 0$ everywhere except the hot wall on the left, and the previously described boundary conditions, the progression of heat through the box is shown in Figure 5.3. All parameters remain the same as those in Figure 5.3, with the exception of the time at which the images were taken. The dimensionless time, t^*, is defined as:

$$t^* = \frac{\alpha_f t_{LB}}{N^2} \tag{5.5}$$

where t_{LB} is the current time step within the lattice Boltzmann simulation.

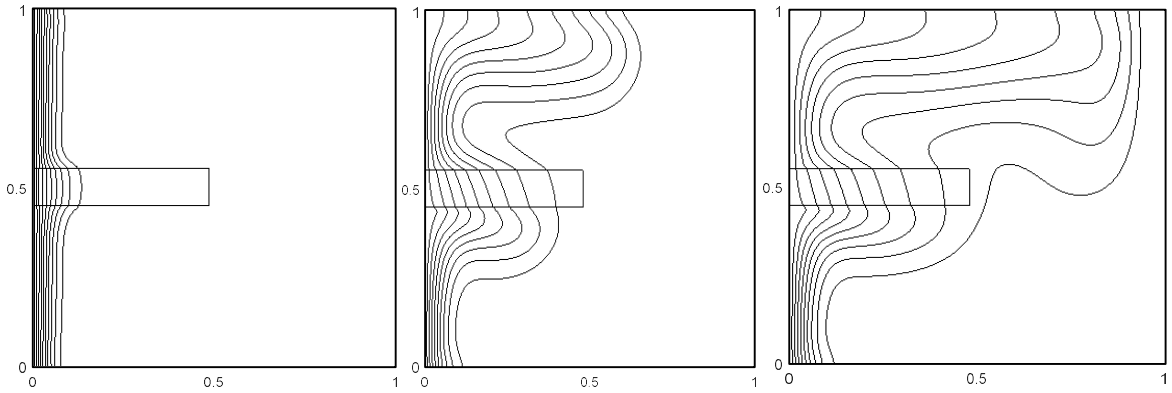


Figure 5.4. Dynamic temperature isotherms for the present work. Images are $t^* = 0.0018$, $t^* = 0.0188$ and $t^* = 0.0375$ respectively.

As expected, at the beginning of the simulations in Figure 5.4, the temperature profile is dominated by heat conduction through the solid, due to the thermal conductivity ratio $\lambda = 10$. Over time, fluid begins to move as it is heated and made less dense, carrying energy with it to the other side of the box.

During the early stages of the simulation, heat in the current work spreads slower, resulting in a difference of approximately one isotherm line. Later in the simulation, the present work appears to catch up somewhat, likely due to the apparently higher velocity increasing the rate of heat transfer from the wall. Over greater time, each simulation approaches a similar steady state. Although most relevant physical systems will have materials with different heat capacitances, in this work the heat capacitance ratio between the phases has been left at unity so that works can be accurately compared.

Similar to the method also used by Frederick and Moraga (2007), the surface averaged Nusselt number Nu_{surf} is calculated to both show the differences between the modes of heat transfer present at varying Ra, as well as give a quantitative measure of the progression of the simulation. The Nusselt number in Equation 5.6 will always start high for this system, before decaying over time as the temperature differential reduces in magnitude.

$$Nu_{surf} = \sum_{z=1}^{N} \sum_{y=1}^{N} \left(\frac{k_{yz}}{k_f} \frac{\Delta T_{yz}}{\Delta x_{yz}} \right) \tag{5.6}$$

Here, ΔT_{yz} is the change in dimensionless temperature between the cells on the hot wall, situated at $x = 1$, and those cells immediately adjacent to them on the lattice slice at $x = 2$. Δx_{yz} is the dimensionless distance between the aforementioned cells and k_{yz} is their thermal conductivity. For the fin geometry of Figure 5.1, the ratio k_{yz}/k_f will be equal to $\lambda = 10$ only

for the solid phase cells within the fin, and is equal to unity for the fluid on the bare wall. Once the relative change in Nu_{surf} between successive steps became small, the system was assumed to have reached a steady state.

The surface averaged Nusselt number tracks well with those found by Imani along the entire simulation time, as shown in Figure 5.5. Although the surface average Nusselt number calculated in the present work is close to that found in the previous work for the entire time domain, a small but consistent offset is present. This offset exists as a result of the slightly higher velocities in the current work, as supported by the small differences in the temperature profiles.

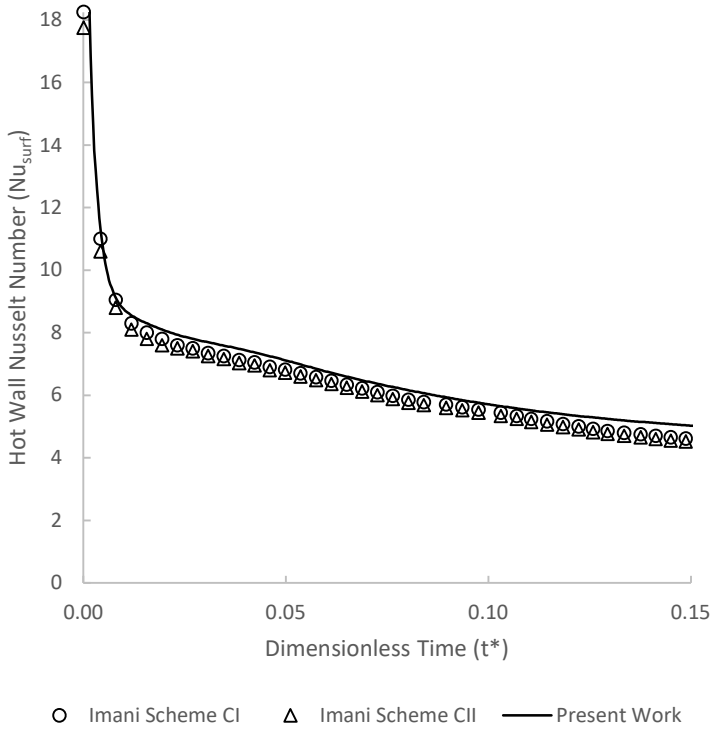


Figure 5.5. Hot wall Nusselt number over time for (line) present work and (O and Δ) Reproduced from Imani (2017) with permission from Elsevier.

In Imani's work, the circle and delta markers are for simulations first without, and then with special treatment of the conjugate boundary, for systems with equal heat capacitance ratios. In this case, the results should be similar, as no special treatment effectively forces the heat capacitances to be equal. The present work made use of the conjugate heat transfer scheme outlined by Lu *et al.* (2017), and detailed in Chapter 4.

5.3 Mesh independence

A mesh independence test was carried out to ensure that the resolution that the simulations were carried out at was appropriate for the parameters in use. In a such a study, the same system is tested across a number of resolutions, and the end results compared. Although in general, higher resolution values tend to provide greater accuracy, this increase comes with diminishing returns. In contrast, the computational time is proportional to the cube of the resolution, before encountering additional issues from memory and disk space limitations at high values. Results were obtained across a number of resolutions and an acceptable resolution chosen as the point where the deviation in the final calculated heat transfer coefficient result is less than 10%.

The heat transfer coefficient, h may be calculated by the method that will be described in Section 6.2. Figure 5.6 gives the result that altering the resolution of the simulation has on the calculated heat transfer coefficient, h.

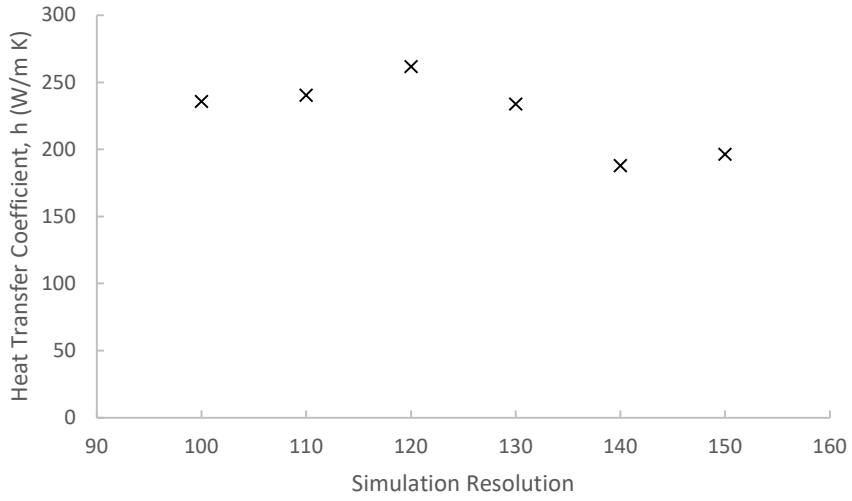


Figure 5.6. Effects of increasing the simulation resolution.

Deviations of approximately 10% from the average result may be incurred depending on the resolution chosen. This is acknowledged as a potential source of error. The resolution used throughout this work is $N = 120$. This resolution was chosen as it provided a middle-ground between additional stability for simulating slightly higher Reynolds, up to approximately $Re = 3$, while being small enough that the program could still be run on desktop computers. The ratio between the thermal diffusivities of the stainless steel solid phase and the water fluid phase, λ is limited to $\lambda = 30$. Outside of this range, the simulations diverged. Although fluid temperature was not a significant factor to the simulation stability alone, relatively high values

were used to maximise the thermal diffusivity of the water, reducing the thermal diffusivity ratio.

The major reasons for the changes in the heat transfer coefficient are simulation stability, and geometric accuracy. The tests at low resolution values tend to become numerically unstable due to the coupling of the hydrodynamic and thermal lattices. In these cases, the simulations diverged, causing non-physical results such as temperatures rising above those of the hot inlet, or below those of the cold, eventually to the point of $\pm\infty$. Although the hydrodynamic lattice tends to exhibit high stability across a wide range of parameter values, the thermal lattice does not. The large variation in thermal diffusivities between the fluid and solid components of the structures further diminishes this.

In this case, a second, more important factor is the translation of the STL file back into a voxel field. The voxelisation algorithm always leaves the outermost layers of cells within the domain as fluids. This occurs even if solids from the STL file were defined in that region. As a result, some truncation of the geometries can occur, as these outermost cells must be removed to seal up the fluid inlet and outlet channels. This inconsistency in the discretisation of the final geometry across simulation resolutions is what causes the small amount of variation in final results calculated, as the structure is represented differently within the simulation voxel fields. Resolutions less than $N = 100$ were not possible, as insufficient voxels were available to ensure the integrity of the solid structures, creating fluid leaks.

6 Performance Investigation

6.1 Overview

The computational models developed within Chapter 4 were used to investigate the parameters used in the design of heat exchangers created through the methods outlined in Chapter 3. While the results generated within Chapter 5 are compared with experiments, this chapter will focus on numerical results only. Within each of the simulations, two primary properties are of interest. These are the pressure drop across each of the fluid channels, and the heat transfer effectiveness of the designs. This effectiveness is measured by the temperatures at each of the entrances, from which heat transfer coefficients may be calculated. From these properties, comparisons may be made between the relative performance of designs utilising different geometries, in order to optimise the designs of the TPMS heat exchangers.

As described in Chapter 2, heat exchangers are commonly evaluated according to their overall heat transfer coefficient U, the number of effective transfer units NTU, and the overall heat exchanger effectiveness ϵ_u. These metrics are dependent on both the structures studied, and the flow profile of the fluids moving through them. The effectiveness is a measure of the relative heat that was transferred, compared with the maximum that was possible for an exchanger of infinite length. Each of these measures must be analysed in conjunction with the pressure drop ΔP across the design, as this determines the number of unit cells that may placed within a structure before the pressure drop becomes excessive. Of interest are designs with high heat transfer coefficient, regardless of pressure drop, for applications where pressure challenges may be easily overcome with pumping, as well as those designs with high heat transfer coefficient to pressure drop ratios, where the most efficient units are desired.

The major parameters fundamentally affecting the spread of heat through the heat exchanger are the thermal conductivity k, the heat capacity c_p, and the density ρ. These properties make up the thermal diffusivity α, which is used as one of the main variables of the thermal simulations. The thermal conductivity controls the rate at which heat is spread throughout the heat exchanger, allowing more conductive materials to conduct heat quickly. The product of the heat capacity and the density, the heat capacitance, provides thermal inertia. Heat capacitance alters heat transfer indirectly through differences in the unsteady temperature differential. The other main variable for the simulations is the Reynolds number Re. The Reynolds number, as defined by the velocity through the channels determines the rates of

advection and convection that may be attained throughout the structure, which make up a large proportion of the total heat transfer.

Testing of the various parameters impacting the performance of the TPMS heat exchangers was completed by choosing one base design and simulating the effects of changing a single parameter at a time. The basic workflow of this process is outlined in Appendix D. The base design is not intended to be the most efficient heat exchanger, but has similarities to those tested experimentally by others (Symon, 2017), while providing faster and more stable computation than higher velocity alternatives. Images of resin printed heat exchangers using these dimensions may be found in Appendix C. Table 6.1 outlines the relevant design specifications for the base design of the TPMS heat exchanger.

Table 6.1. Specifications for the base TPMS design.

Property	Value
TPMS type	Gyroid
Flow configuration	Counter-flow
Solid material	Stainless Steel
Fluid material	Water
Porosity (%)	75.2
Length unit cell (mm)	10
Channel diameter (mm)	4
Internal wall thickness (mm)	1
External wall thickness (mm)	3
Re	1
Pr	2.39
Simulation resolution	120
$T_{C,in}$ (°C)	70
$T_{H,in}$ (°C)	80
Length of packing (mm)	109
Tube packing I.D. (mm)	34

The LBM model used is capable of capturing the unsteady behaviour of the heat exchanger systems, as described in Chapter 5. For the purposes of the calculation of most of the quantities found within this chapter, only the steady state behaviour is considered. In these cases, the simulations are allowed to run until a steady state has been achieved, as according to the convergence criteria:

$$\epsilon_{tol} \leq 10^{-5} n_{save} \tag{6.1}$$

The n_{save} term is the period at which simulation data is written to disk, and is introduced to compensate for the fact that this quantity is only calculated at each of these intervals, in the

form of a visualisation toolkit image data (VTI) file. Although this delay to the calculation means that the point where the tolerance is met is never precisely identified, overshooting for greater accuracy is only a computation time cost.

Simulations were split between a desktop PC using an Intel i7-4770 quad-core processor and 16 GB of memory, and a server using two Intel Xeon X5670 hexa-core processors, with 96 GB of memory. Parallel operation was enacted through the use of the MPICH implementation of the Message Passing Interface (MPI). By this method, the simulation domain was split up between each of the available cores, allowing for faster total computation time for machines with a greater number of cores. The use of the desktop PC was restricted from higher resolutions by the need to store the entirety of the lattice data in the memory of the machine in use, despite the core parallelisation for the computations.

6.2 Performance metrics

A range of identifying characteristics and performance metrics are investigated to quantify the performance of the heat exchange structures. These quantities are used to measure the pressure drop and heat transfer characteristics of each representative element. The dimensionless quantities calculated may be compared to those found by other authors.

The heat transfer based Péclet number gives the ratio of advective heat transfer to diffusive. For low values, diffusive movements of heat dominate, and the conductive heat transfer of laminar fluid flow is likely. Throughout this work, the Prandtl number is maintained constant, as the same fluid, at the same average temperature is used consistently. For this reason, Péclet number will always be directly proportional to the Reynolds number.

$$Pe = \frac{L_{char} u}{\alpha} = Re.Pr \tag{6.2}$$

The next, and most simple quantity is the porosity. This is defined as the ratio of the fluid void volume, V_f, to the total volume of the structure, V. Within the structures investigated in this work, each of the fluid voids will be equal, and the remainder is made up of the cell wall.

$$\phi = \frac{V_f}{V} \tag{6.3}$$

Porosities are denoted as belonging to either channel A, ϕ_A, or channel B, ϕ_B. Each of these will be equal to less than $\phi \leq 0.5$. Porosities vary according to the wall thickness and the specific TPMS structure studied.

Tortuosity is defined for a given unit cell as the ratio of the length of the mean winding free space, L_{free}, to the length of the entire cell, L, as demonstrated by Equation 6.4. This ratio will always be greater than, or equal to unity, with the latter occurring only within the most simple, unobstructed paths.

$$\mathrm{T}_{straight} = \frac{L_{free}}{L} \tag{6.4}$$

Although shorter paths through the structure may be possible, the tortuosity is calculated using the midpoint of the channels. However, for complex structures containing multiple channels of different lengths, which may or may not be interconnected, the definition in Equation 6.4 is insufficient. An alternative form shown to be effective for the measurement of porous media (Matyka & Koza, 2012) is given by:

$$\mathrm{T} = \frac{\langle u \rangle}{\langle u_z \rangle} \tag{6.5}$$

The quantity $\langle u \rangle$ is the average velocity magnitude within the flow structure, while $\langle u_z \rangle$ is the component of the velocity along the primary flow direction. These quantities were evaluated at each lattice fluid location within the structure, averaged together, and written to file at steady state. At this steady state, the velocity distribution will take on the same distribution of physical path lengths throughout the structure. Figure 6.1 demonstrates an example of the paths that may be taken through an example structure. Arrows indicate the fluid movement, while the shaded region is the solid phase, ϕ_s. More tortuous paths will generally incur higher pressure drops. The heat transfer properties of the structure however are more closely linked to the individual shapes being used, and so cannot be predicted by tortuosity alone.

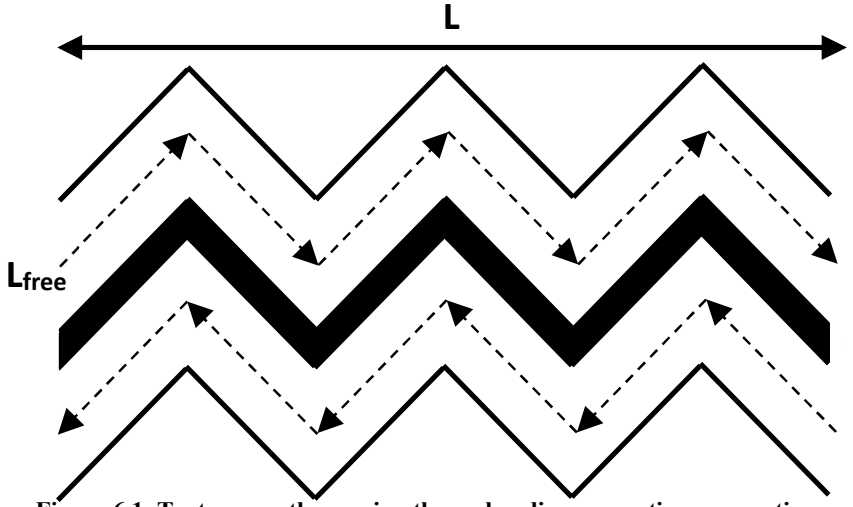


Figure 6.1. Tortuous paths moving through a diagrammatic cross-section.

The permeability, K, as found by Darcy's Law, Equation 6.6, describes the ability of a porous medium to transfer fluid through itself. It depends on the kinematic viscosity ν_f, the superficial velocity u_s, and the pressure drop per unit length over the structure, $\frac{\Delta P}{L}$. High values of permeability indicate that faster fluid movement will be possible for a set pressure drop. Applying this to usage within heat exchangers, higher permeabilities allow operation with lower pumping costs.

Although the permeability depends on the superficial velocity passing through a structure, for basic TPMS designs containing identical fluid channels, permeability within a TPMS structure will be equal for each channel (Jung & Torquato, 2005).

$$K = \frac{u_s \mu_f}{\frac{\Delta P}{L}} \tag{6.6}$$

To compare to other works, the permeability must be non-dimensionalised, such that it is no longer dependent on the length scale in use. This is achieved by dividing by the square of the characteristic length. One definition of the Darcy number, Da, also sometimes known as the dimensionless permeability, may then be compared to those found by other authors for TPMS structures.

$$Da = \frac{K}{L_{char}^2} \tag{6.7}$$

While the permeability and Darcy number measure the effect the pressure drop has on the flow through a porous medium, the Bejan number provides a means of non-dimensionalising the pressure drop for comparisons. The Bejan number dimensionless grouping (Bejan, 2004; Petrescu, 1994) is used within low Reynolds applications.

$$Be = \frac{\Delta P L^2}{\mu \nu} \tag{6.8}$$

For comparisons within the current work, where the same flow rate is usually in use, Bejan number comparison is possible. However, Equation 6.8 shows that Bejan number is proportional to the length scale in use. Therefore, an alternative comparison must be made. As shown by Houlton (2019) for a selection of unit cells of various designs, the Bejan number is linearly proportional to Péclet number. As such, the ratio of the dimensionless pressure drop, the Bejan number, to the dimensionless heat flow rate, the Péclet number, will be a constant, Be', for each structure investigated. Approximately linear trends can be expected when this

ratio is investigated. Higher values of this constant indicate structures where the flow rate has a greater effect on the pressure.

$$\frac{Be}{Pe} = Be' \tag{6.9}$$

In the study of porous media, heat transfer is sometimes investigated through the assumption of either a constant temperature or heat flux from the solid phase. The temperature change of the fluid at the outlet is then examined. In the case of systems with two distinct fluid channels, heat exchanger theory may instead be used to calculate the heat transfer coefficient. By imagining each representative volume as a full heat exchanger, an overall heat transfer coefficient, U, may be found:

$$U = \frac{Q}{A_s \Delta T_{mean}} \tag{6.10}$$

where A_s is the fluid wall surface area, Q is the net heat transferred between the channels and ΔT_{LMTD} is the log meant temperature difference between the channels. The wall heat transfer coefficient h, is found by assuming that each side of the TPMS walls were approximately equal in heat transfer capacity, through Equation 6.11. For this assumption to be valid, each of the channel sets must have similar flow conditions, measured through the Reynolds number and flow rate.

$$h = \frac{1}{\frac{1}{2}\left(\frac{1}{U} - \frac{\Delta x_w}{k_s}\right)} \tag{6.11}$$

As discussed in Chapter 2, further metrics, including the Nusselt number, number of equivalent heat transfer units, and effectiveness could be calculated. This value could then be non-dimensionalised through the calculation of the Nusselt number, Nu.

$$Nu = \frac{hD_H}{k_f} \tag{6.12}$$

The Nusselt number is a metric used widely across many different fields to describe the mode of heat transfer. For low values, the ratio is dominated by the thermal conductivity term, and heat transfer is by conduction only. As the Nusselt number increases, the heat transfer coefficient rises, and generally so will the rate at which heat may be transferred. As the Nusselt number is dimensionless, it may be applied across a wide range scales and applications.

6.3 TPMS Morphologies

A number of common TPMS structures were tested, as well an additional structure designed to emulate a shell and plate design. The TPMS structures included, as outlined in Chapter 2 are the Schoen gyroid, Schoen IWP, Schwarz primitive and the Schwarz diamond. Each of these geometries has been proven to be an efficient morphology for applications within other studies. Although each of the TPMS structures tested have similar surface areas per unit cell, creating TPMS STL files of different varieties, while maintaining identical surface areas was deemed impractical. These surface area values are included within the calculation of the heat transfer coefficient, but may influence pressure drops in unexpected ways.

Each of the TPMS designs were tested in the form of individual unit cells that would make up a greater whole. Fluids enter and exit the unit cell within their respective channels, with no physical mixing allowed. One channel is the hot inlet, while the other channel is the cold inlet. Figure 6.2 shows a diagram of an example gyroid unit cell, along with some key dimensions.

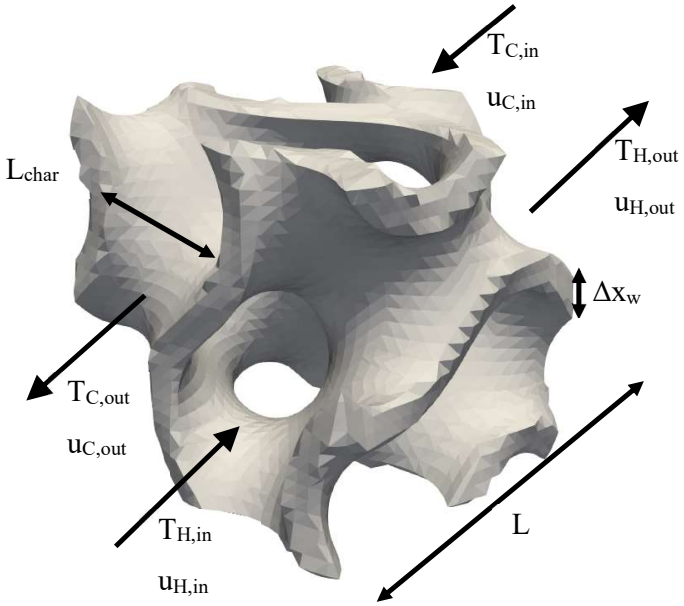


Figure 6.2. Inlets and outlets of an example gyroid unit cell.

In relation to the full heat exchanger structures, these unit cells make up the entirety of the heat transfer channel volume, so any trends observed in the heat transfer performance of these may also be extended to the full designs. Figure 6.3 demonstrates the location of one of the TPMS unit cells within an example full-scale heat exchanger design.

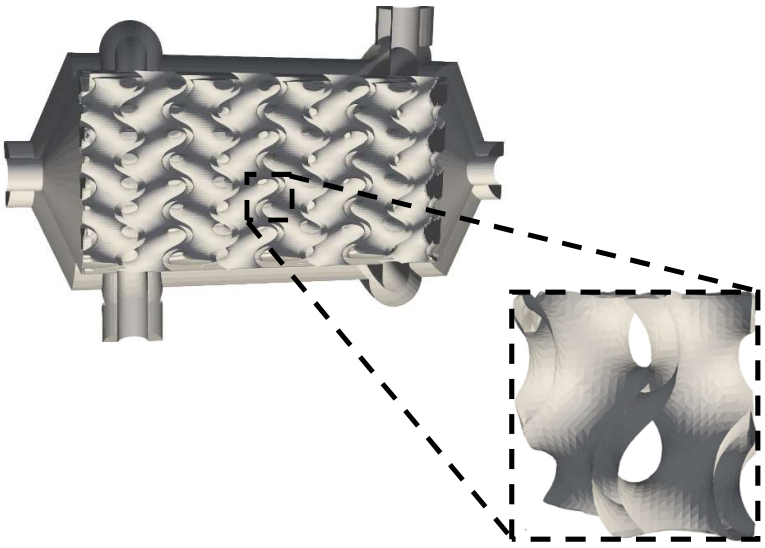


Figure 6.3. Location of a gyroid unit cell within a gyroid based TPMS heat exchanger design.

The first morphology tested is a simple baseline case where flat plates are used in place of the TPMS channels. A unit cell of this design is shown in Figure 6.4. In this case, alterations to the full-scale design were required at the packing edges to ensure that the channels remained properly separated. A connecting tube is sent through each of the plates to link together the plates on the secondary channel set. The flat plate designs are tested for comparison to the TPMS, as although they are expected to have the lowest pressure drop, they are also expected to have amongst the worst heat transfer characteristics. This design most closely resembles a more conventional shell and plate type of heat exchanger.

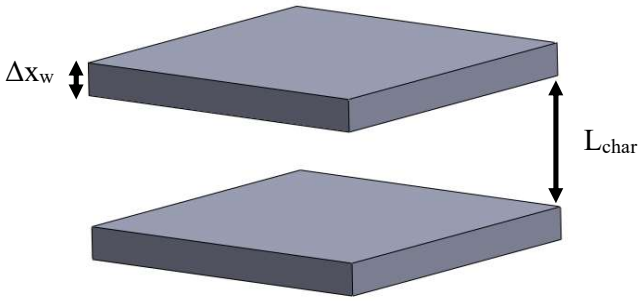


Figure 6.4. Flat plate geometry incorporated into a unit cell.

The majority of the variance in surface area between the structures is due to shape effects from the different morphologies studied. Although they share a common channel diameter, each design has a different relationship between the diameter and the surface area. A minority of the variance is related to the STL generation algorithm itself, which varies according to the STL mesh resolution used. Although the surface area of each TPMS geometry are on the same order

of magnitude, they do differ significantly between designs. Ideally, only the surfaces present in every unit cell would be included, such that those of the solid portion on the outermost wall of the domain be removed for accuracy. However, some deviation from this can exist if the mesh resolution is too low.

The choice of mesh resolution used to create the STL file, as carried out in Section 3.3, was made by considering file size, computation time and the smoothness of the resulting shapes. At some critical threshold, the surface area of a given structure will stop significantly rising with mesh resolution, as a maximum smoothness of the final part is achieved. Figure 6.5 gives the resulting surface area calculated for different mesh resolutions, each for the otherwise same Schoen gyroid unit cell. At a mesh resolution of $N_{mesh} = 30$, the change in surface area was deemed low enough that the part was acceptable. Although higher mesh resolution parts could have been used, the later voxelisation steps, whereby the STL file is converted back into simulation nodes on the cubic LBM grid, would have removed any potential gains from doing so. One potential source of error present at this stage is the difference between the areas utilised by the voxelised fields within the LBM code, where the heat transfer actually takes place, and those of the STL file, which most closely resemble the real surface. Finding the actual surface areas of the cubic voxel fields within the LBM was deemed too difficult, while still not accurately portraying the real surface area normal to the boundary. Usage of the surface area calculated from the STL was deemed the best choice overall, as it represented the physical system the best. Across all these factors, all shapes experienced a similar degree of error, with the exception of the flat plate example, due to the lack of a curved boundary.

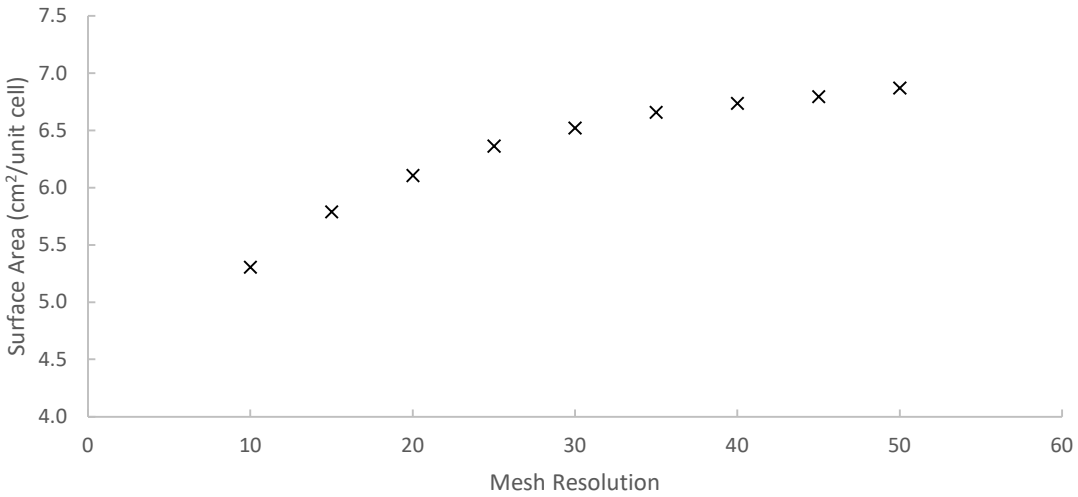


Figure 6.5. Surface area of the STL representation for Schoen gyroid unit cells of varying mesh resolution.

6.4 Parametric Testing

Through alterations to the TPMS designs, the performance of the heat exchange structures may change greatly. Any alteration that alters the flow profile of fluids passing over the walls will have continuing effects on the rate of heat transfer through the walls. The resulting change in equilibrium position for the fluid channels will result in differing outlet temperatures, and therefore, different calculated heat transfer coefficients. One of the most obvious factors impacting the performance of the TPMS is the specific TPMS used. Gyroidal cells move fluids in a different manner to diamond cells, regardless of the scaling criteria to which the geometries are compared.

Quantities that are dependent only on the specific structure studied and are independent of the flow rate are important for identification. These quantities may be compared the most easily between different works. The tortuosity, as described in Equation 6.4, would for a simpler structure be defined as the ratio of the mean curved path distance travelled by an element in the fluid, compared to the equivalent linear distance in the axial direction. This equation contains only distance terms, implying that there should be no effect of flow rate, or correspondingly the Reynolds number, on the tortuosity found. Within Equation 6.5, the numerator and denominator are each velocity terms, for which the scale will cancel. TPMS geometries were tested across a range of flow rates and the results were plotted in Figure 6.6.

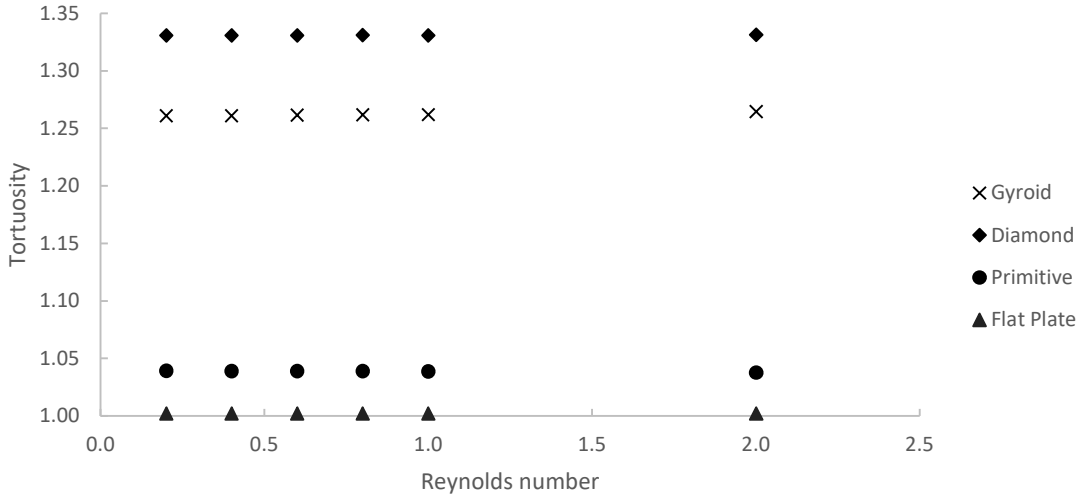


Figure 6.6. Tortuosity of TPMS shapes across a range of Reynolds numbers.

Although a very slight positive trend is present, within the ranges investigated within this work, the tortuosity is independent of the flow rate, and corresponding Reynolds number. Another quantity, the dimensionless permeability, also known as the Darcy number, must also be

checked for flow independence. Each of the major designs were tested across a range of Reynolds numbers. Most of the geometries were unchanged by the increasing Reynolds number, showing they are independent. The primitive geometry that was tested does show a small reliance on the Reynolds number, and may not be as easily comparable to other works as a result. Figure 6.7 shows these results.

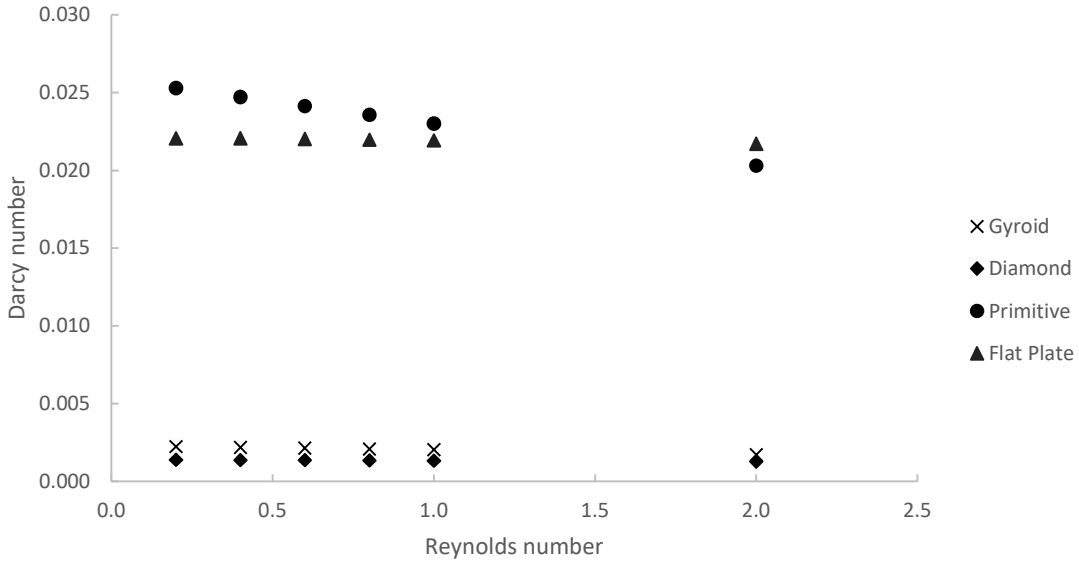


Figure 6.7. Darcy number of a Schoen gyroid, across a range of Reynolds numbers.

Another parameter requiring comparison for independence is the length scale used within the representative elemental volumes simulated. As the size of the unit cell was altered, the relative channel diameter and wall thickness were maintained the same. Each of the changes in the channel diameter are offset by a corresponding change in the fluid velocity, such that the Reynolds number is set to $Re = 1$. As the calculated Nusselt numbers are all very similar, across a wide range of cell lengths, the two quantities may be assumed to be independent. Figure 6.8 shows this lack of obvious correlation.

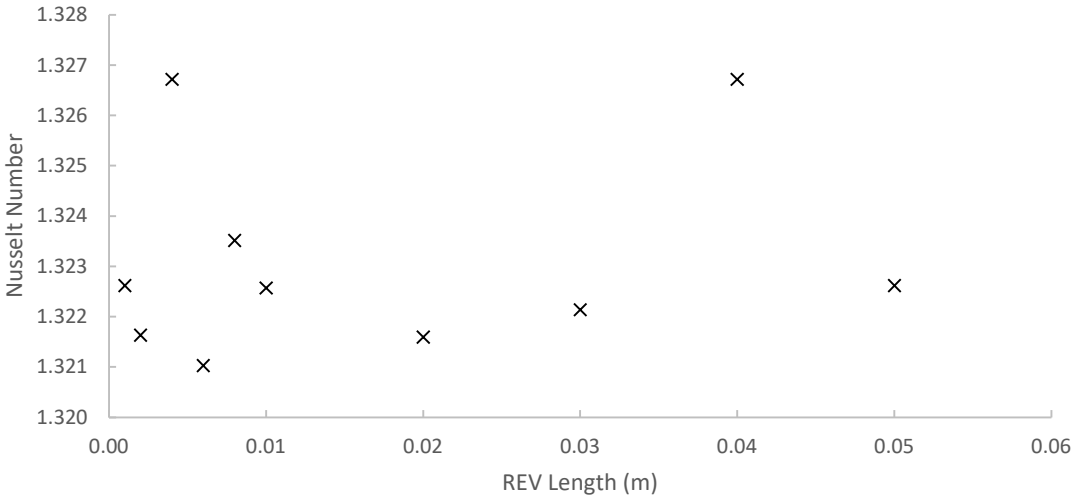


Figure 6.8. Effect of REV size on the Nusselt number found for a Schoen gyroid structure.

Each of the generated TPMS structures are capable of different performance depending on the orientation at which the TPMS tubes are inserted into the shell. On some orientations, a direct fluid path may pass directly through the shell, while others may necessitate a zig-zag type of flow path, as the direct path is always blocked. While direct paths will likely yield lower pressure drops, heat transfer will also be hampered. Each of the TPMS were trialled along a subset of orientations, whereby each of the primary x and y axes were rotated by 45°. Figure 6.9 demonstrates the Nusselt numbers found for a selection of orientations, for each of the gyroid, diamond and primitive structures.

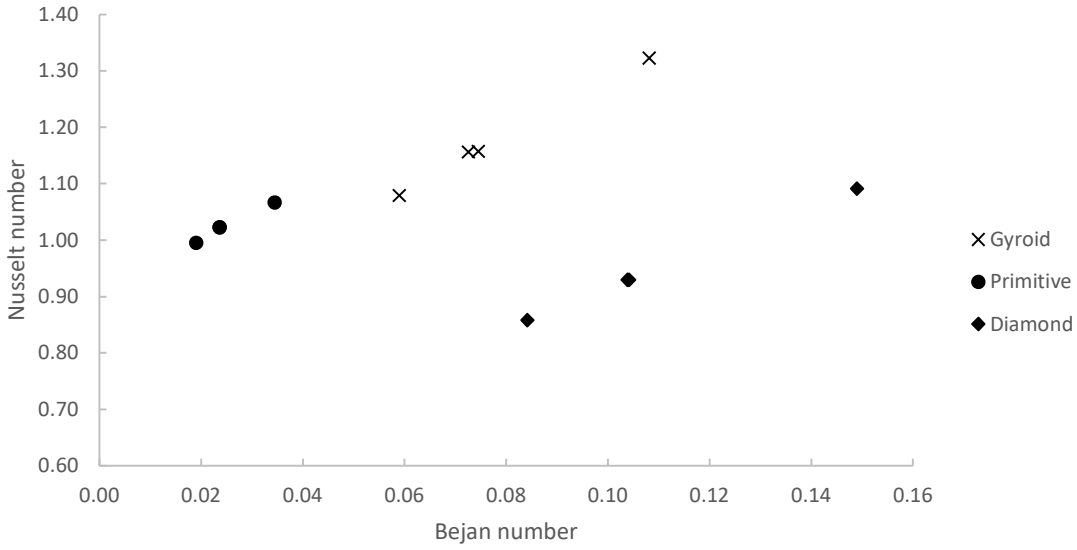


Figure 6.9. Performance of each TPMS design with altered orientation.

The orientation changes are described through the use of Miller indices. Miller indices describe the orientation of a unit cell, and the grant a method to easily relate rotations in three dimensions. Fluids of the primary flow channel enter the unit cells along the (0,0,1) direction. Each of the orientations tested rotate the cell by 45°, along each of the x and y axes. For this reason, the alternative orientations trialled were the (1,0,1) and (0,1,1) directions. The combination of these two directions was also considered, giving the final (1,1,1) orientation.

To actually test the different orientations, instead of reorienting the unit cell, the flow direction at the inlets was altered by an equal amount. The presence of periodic boundaries on the non-primary edges of the domain meant that the TPMS structure was still traversed in the same way by the fluid, as if the domain itself had been moved. Doing so helped prevent potential errors caused by the generation of different STL files to form the domain. The magnitude of the velocity always remained the same, but was split into multiple directions instead of entering entirely on a single cardinal direction.

The orientations of the TPMS unit cells in Figure 6.9 are found to make a difference, but not significantly more so than the variation between the TPMS designs. In each case, the default (0,0,1) orientation was found to give the best results, the (1,0,1) and (0,1,1) orientations were almost identical, and the (1,1,1) orientation gave the lowest Nusselt number.

For each of the TPMS structures, the relationship between the tortuosity, dimensionless pressure drop and Nusselt number are examined. The ratio of the Nusselt number to the tortuosity is shown for a range of TPMS structures in Figure 6.10. Although each of the TPMS geometries are grouped together by their respective tortuosity values, the Nusselt number of each orientation within the grouping varies.

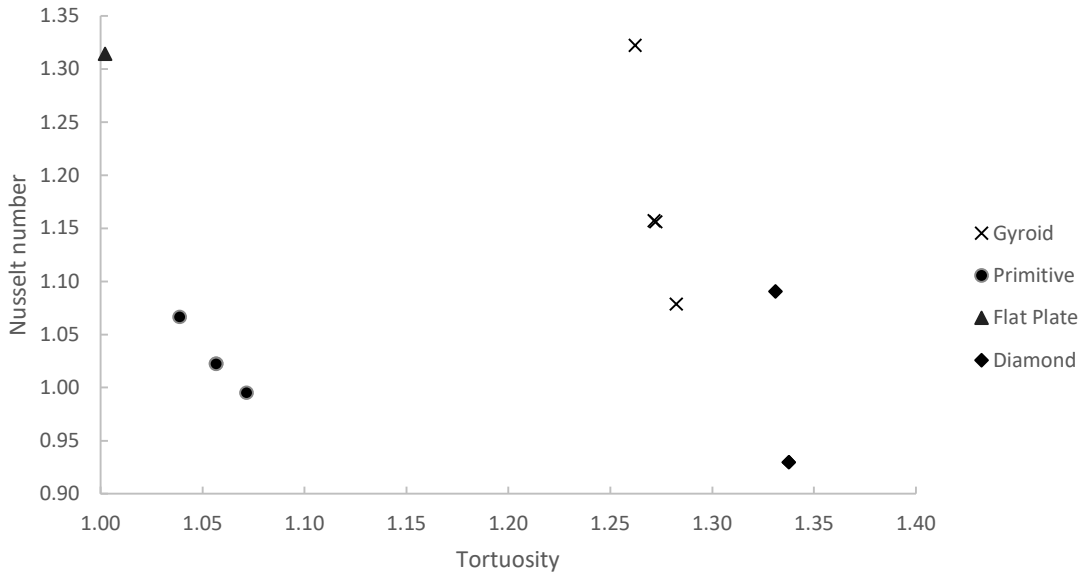


Figure 6.10. Effects of the tortuosity on the Nusselt number.

Figure 6.10 demonstrates that tortuosity alone is an insufficient metric for predicting the performance of a heat exchanger system. The particular Nusselt number found varies greatly depending on the internal structure in use. The flat plate baseline case has very low tortuosity, while also maintaining one of the highest Nusselt numbers. The gyroid structures provide only a slight increase in heat transfer, while experiencing significantly higher tortuosity.

Although the tortuosity cannot be directly correlated to the Nusselt number, it may be compared with the Bejan number. The tortuosity of the flow is expected to be the largest contributing factor to the size of the pressure drop across a unit cell. The higher the tortuosity, the larger the effective surface area that must be passed over by the fluid between the cell entrances. Figure 6.11 shows this correlation, across a number of cell orientations. In the case of the flat plate, only the most simple, perpendicular orientation was tested.

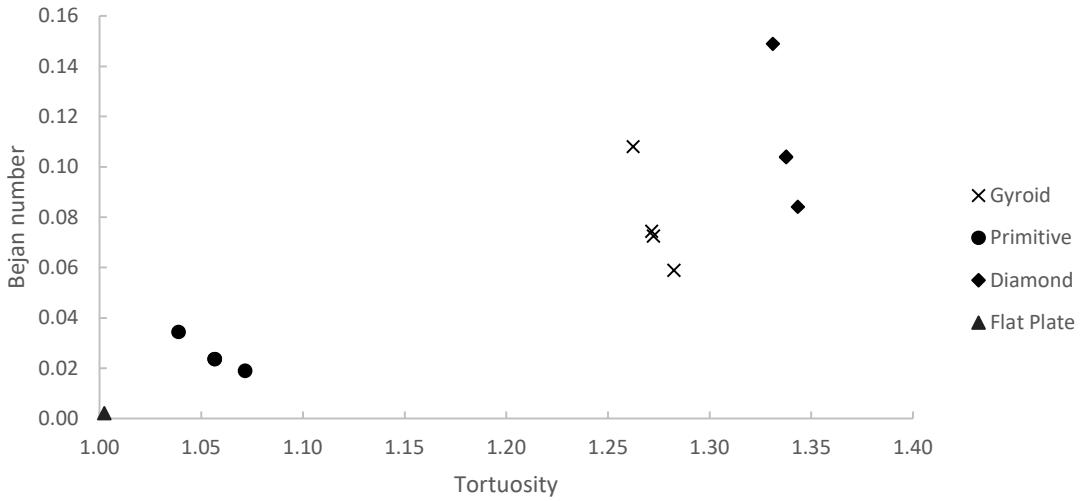


Figure 6.11. Effects of the tortuosity on the Bejan number.

Figure 6.11 shows that although variance is present according to the orientation of the unit cell, a linear correlation appears to exist between the tortuosity and the dimensionless pressure drop. For low Reynolds number flows, it may be possible to approximate the pressure drop across a TPMS structure without simulation being required.

By dividing the Nusselt number from Figure 6.10 by the Bejan number from Figure 6.11, a dimensionless ratio analogous to the heat transfer per unit pressure drop may be found. Figure 6.12, prepared at $Re = 1$, shows the orientations of each of the TPMS designs grouped together, with a clear distinction present between them. The flat plate design, owing to the very low pressure drop it incurs, exhibits the highest of this metric.

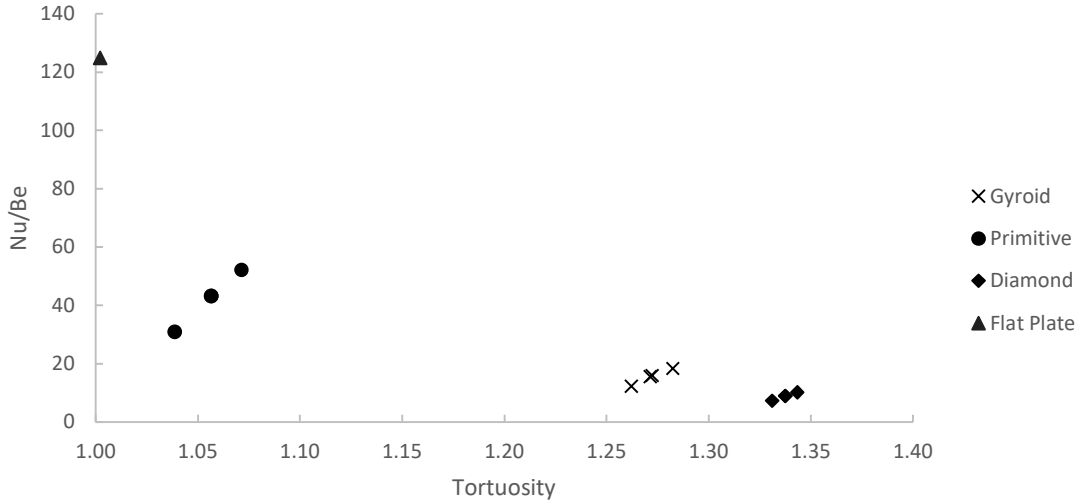


Figure 6.12. Effect of the tortuosity on the ratio of Nusselt number to Bejan number.

The pressure drop across a heat exchanger is a fundamental determining factor in the design of new units. Prediction of the pressure drop for an individual TPMS geometry, across a range of channel diameters, and flow rates may be possible through analogy to porous media. If the TPMS channel structures may be considered a form of consolidated porous media, the internal channel systems of the TPMS heat exchanger may be imagined as a solid, interspersed with connected porous channels. Figure 6.13 gives the pressure drop across a number of gyroidal structures with different channel diameters. Each unit cell maintains the same STL structure but is based on a different length scale for the full cell. The Reynolds number, and other flow parameters are maintained constant for each.

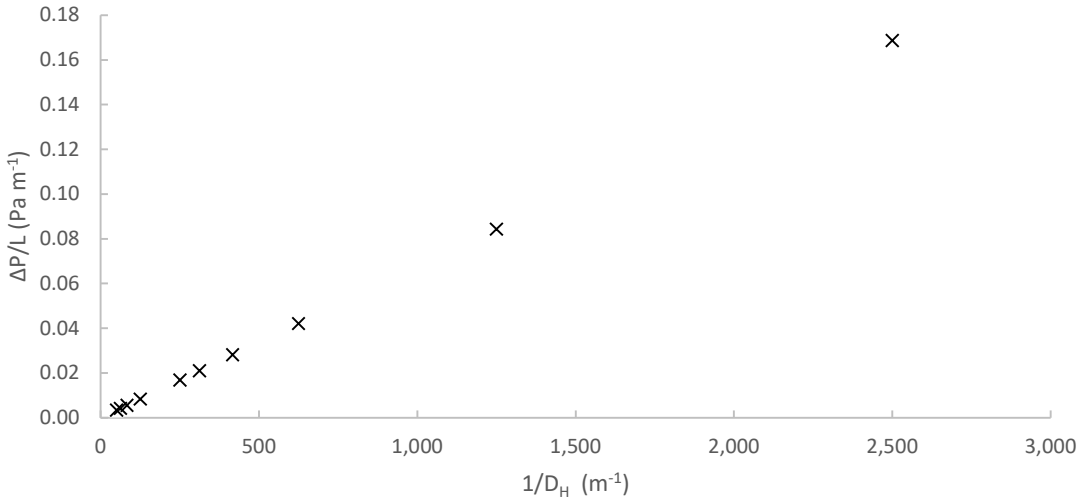


Figure 6.13. Reliance of the pressure drop on channel diameter for a range of gyroid structures.

Since the pressure drop appears to correlate linearly with both the channel diameter and the tortuosity of the structure, basic TPMS geometries should be predictable. These correlations are likely valid only for the laminar flow present at the low Reynolds numbers tested.

Another factor expected to significantly impact the performance of the heat exchanger structures is the relative flow direction of each channel. Typically, parallel-flow operation offers lower heat transfer ability compared to counter-flow alternatives. Each of the designs are next tested with the inlets to each channel system situated on the same side of the unit cell, with otherwise identical flow conditions. Counter-current designs generally grant increased performance compared with the equivalent parallel-flow operation, due to the higher average temperature difference throughout the heat exchanger structure. Figure 6.14 gives the Nusselt number for the base design of each of the TPMS structures, compared with the dimensionless pressure drop, as defined by the Bejan number.

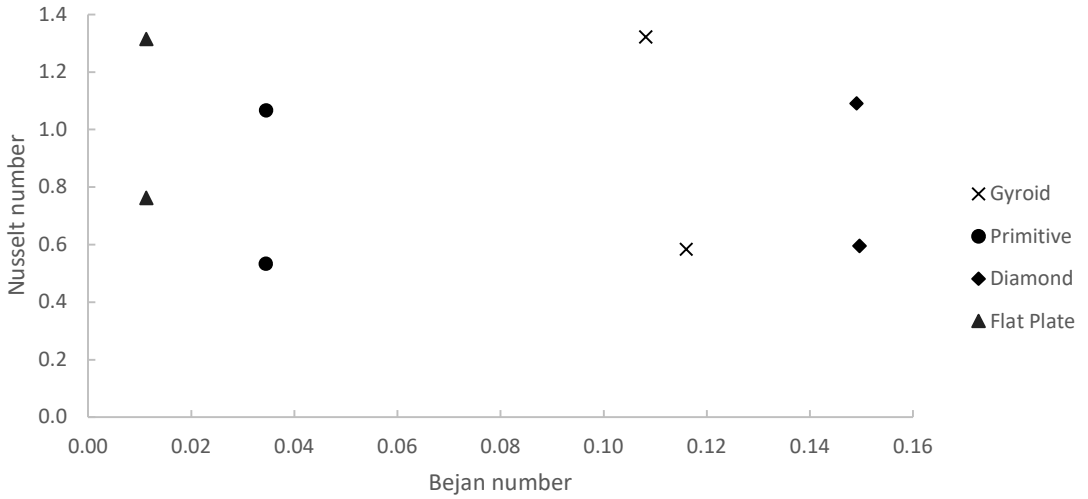


Figure 6.14. Comparison between parallel-current and counter-current operation for each TPMS design.

When operated under a parallel-flow situation, each of the designs performs significantly worse than the equivalent counter-flow test, under identical conditions. Parallel-flow consistently operated at a difference of approximately $Nu = 0.5$ lower than the counter-flow tests. The pressure drop, and the associated dimensionless groupings, are almost identical for each.

The primary means of measuring the rate of heat transfer through the heat exchanger structurers, across a range of flow rates is through the Nusselt number. For very low Reynolds number, it has been that identical structures will often have an approximately linear relationship between the Nusselt and Reynolds numbers shown (Celik *et al.*, 2017; Kuwahara *et al.*, 2001). Individual geometries however will vary, and moving out of the laminar flow regime will cause non-linearity in the relationship. Although other TPMS may also be expected to maintain linear relationships between the Reynolds and Nusselt numbers, each design is expected to fit on an individual and unique curve.

A number of empirical relations developed by various authors for a range of porous media were trialled for comparison to the present results. The relations trialled include Equations 6.13 and 6.14 by Kuwahara *et al.* (2001), which were measured through a porous media consisting of an array of squares, across a range of Reynolds numbers.

$$Nu = 1 + \frac{4(1-\phi)}{\phi} + \frac{1}{2}(1-\phi)^{\frac{1}{2}} Re^{0.6} Pr^{\frac{1}{3}} \tag{6.13}$$

$$Nu = 0.124 \left(\frac{3\pi\phi}{4(1-\phi)}\right)^{0.605} (Re\ \mathrm{Pr})^{0.791} \tag{6.14}$$

Kamiuto and Yee (2005) derived their Nusselt number relationship by examining experimental data from a range of sources, each of which studied different porous media. Theirs is Equation 6.15:

$$Nu = 0.124(Re\ Pr)^{0.791} \tag{6.15}$$

Nakayama *et al.* (2009) carried out heat transfer simulations of both consolidated and unconsolidated porous media. They compared their work to other empirical equations, and formed their own. Equation 6.16 gives their relation:

$$Nu = 0.07\left(\frac{\phi}{1-\phi}\right)^{\frac{2}{3}} Re\ Pr \tag{6.16}$$

Wakao and Kaguei (1982) developed their expression for packed beds of particles. Equation 6.17 gives the final expression compared:

$$Nu = 2 + 1.1Re^{0.6}Pr^{\frac{1}{3}} \tag{6.17}$$

Figure 6.15 shows each of these relations together, along with the gyroid from the present work.

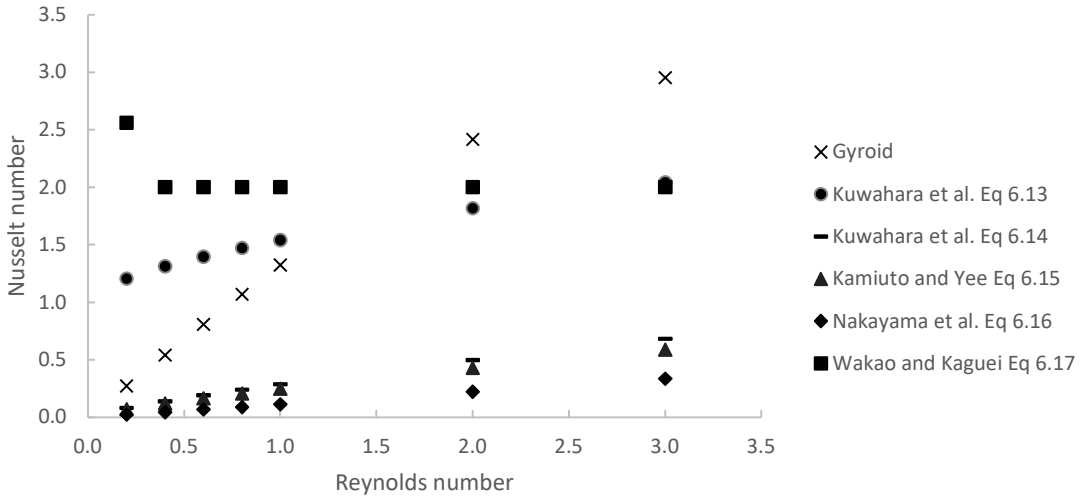


Figure 6.15. Nusselt numbers of various expressions from literature, compared with the gyroid from the present work.

The data from the present work is placed approximately in the middle of the various empirical correlations trialled for the calculation of the Nusselt number. The porous media from each of the studies varied, with some as simple as squares on a grid, while others derived from tortuous channels through a structure. Although the current work falls between many of the other expressions, there is not a good agreement with any specific equation tested. The gradient of

the curve from the gyroid data suggests a greater dependence of the Nusselt number on the Reynolds number than the equations tested.

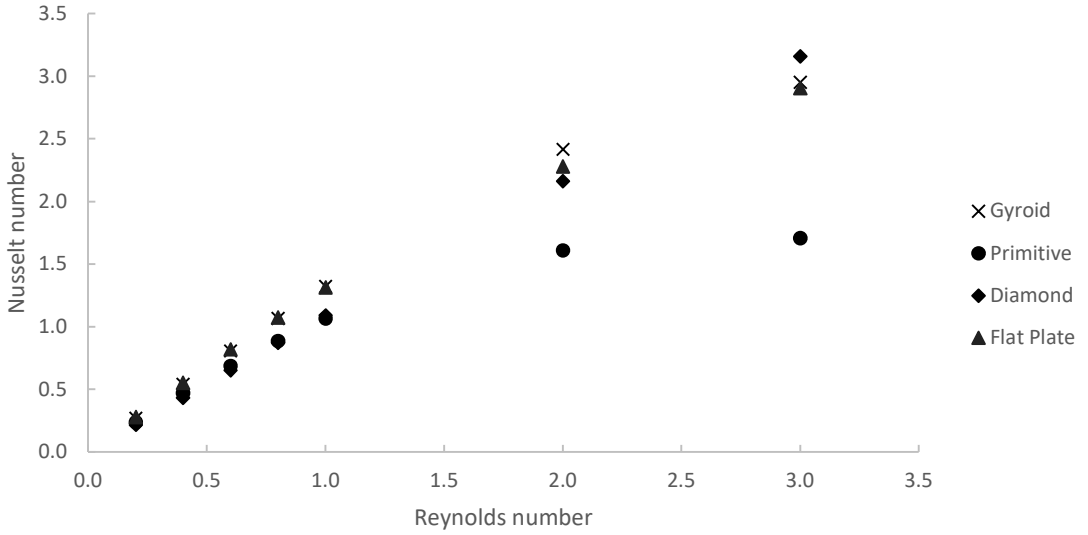


Figure 6.16. Comparison of Nusselt number for different TPMS units.

Figure 6.16 shows that the linearity of the Nusselt number to Reynolds number relationship for the primitive structure begins to break down at the upper end of the range tested. This non-linearity may be related to the flow dependence of the Darcy number that was observable in Figure 6.7. Since the primitive is one structure where the cross-sectional area of the two channels systems may vary greatly, numerical instability was likely encountered after the value of $Re = 1$ to which the simulations were based, when one channel had a large localised deviation in velocity from the mean. As such, the primitive results are likely only valid up to the maximum of $Re = 1$.

Of those tested, the gyroid exhibits a slightly higher Nusselt number than the other designs. However, the flat plate example performs only slightly worse across the entire range. As the flat plate minimised tortuosity and exposed surface area, it may be inferred that for heat exchange structures at low Reynolds and Péclet number, the choice of structure is relatively unimportant, due to the dominance of diffusive heat transfer. Figure 6.16 shows that despite the high Nusselt numbers obtained, gyroidal structures also tend to have amongst the highest Bejan numbers. Notably much lower in pressure drop than the remaining structures, the primitive shows potential for applications where low pumping is available.

Also of interest is the relationship between the Nusselt number, proportional to the rate of heat transfer, and the Bejan number, proportional to the pressure drop. The gradient of the Bejan

number, is unique for each structure studied. Figure 6.17 demonstrates the relationship between Bejan number and Nusselt number for each of the TPMS designs.

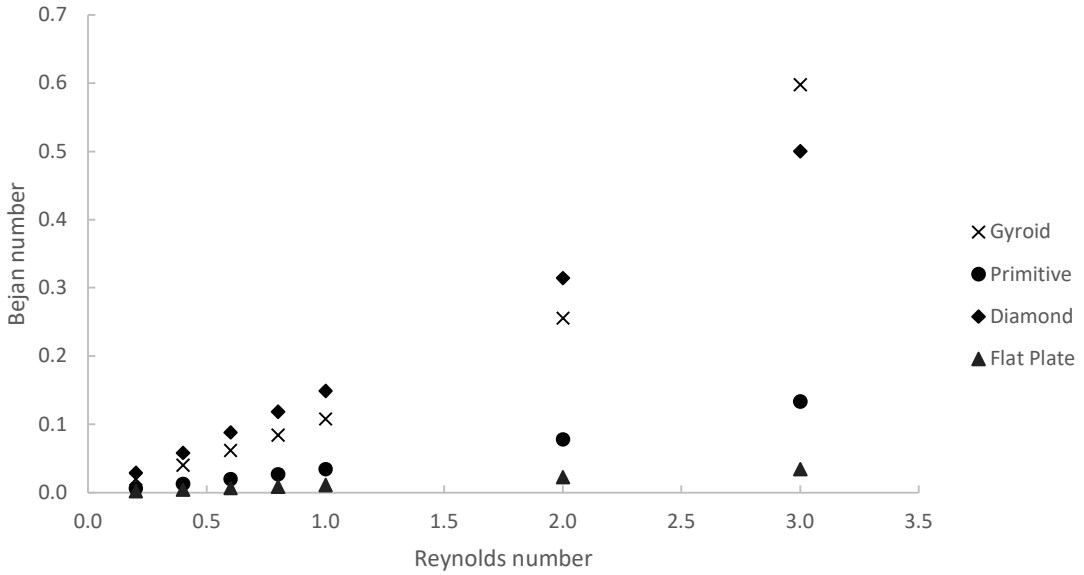


Figure 6.17. Comparison of Bejan number for different TPMS units.

Figure 6.18 shows the relationship between the Nusselt number and Bejan number for each design, where each data point represents one of the previously tested Reynolds numbers. As the gradient of the curve for the flat plate geometries is the greatest, this implies that these structures most efficiently transform fluid movement into heat transferred, for low Reynolds number flow. In general, at low Reynolds numbers, the pressure drop appears to be more strongly affected than the rate of heat transfer by the tortuosity and other shape effects of the TPMS. The most efficient TPMS was the Schwarz primitive, again due to the low tortuosity, and thus reduced pressure drop across the structure. In applications where the pressure drop is unimportant, the Schoen gyroid performed slightly better than the alternatives tested. Other untested TPMS could perform better still.

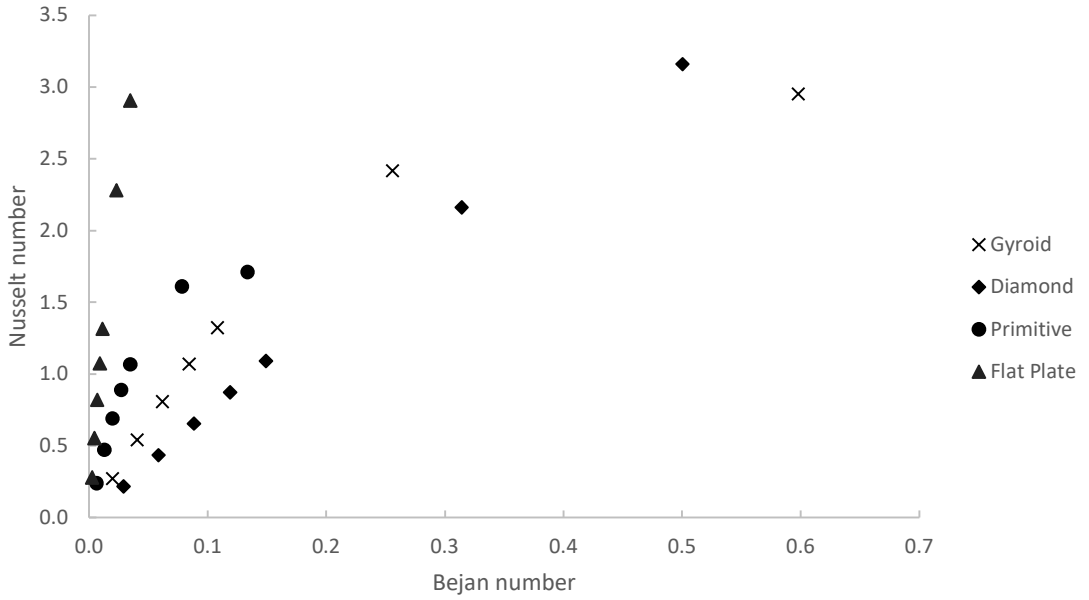


Figure 6.18. Comparison of Nusselt number to Bejan number for each TPMS design, across a range of Reynolds numbers.

6.5 Full-scale heat exchanger structures

The full-scale version of the TPMS simulation program was used only for examining the velocity profile of the full-scale heat exchangers formed. The calculation of thermal effects by this method was found to lack numerical stability without the use of prohibitively large simulation resolutions, which were impossible to achieve using the machines available. Even if making use of supercomputing resources, excessive computational time would still be encountered due to the need for adjacent simulation nodes to communicate.

Along the axial direction, fluids enter through the inlet piping, and reach the TPMS tubing. While the central axis is preferred by the fluid, some flow does move out to the edges of the shell. The fluids move through the TPMS units and exit out the opposite side. On the non-primary direction, the fluids enter through the piping, before entering the flow distributor stretched around the exterior of the shell. A direct path into the TPMS tubing may not always be guaranteed, depending on the specific TPMS, but an entrance at some point along the flow distributor circumference is very likely. The fluids move back in a counter-current manner, exiting through the outlet on the opposite side, again using the other flow distributor if necessary. An iso-velocity surface with a limit of $u = 0.01u_{max}$ for a full-size gyroid heat exchanger is shown in Figure 6.19. The fluid does move through the central channels, but not

all of the outer channels connect properly. This indicates that the flow distribution to the edges of the unit is poor.

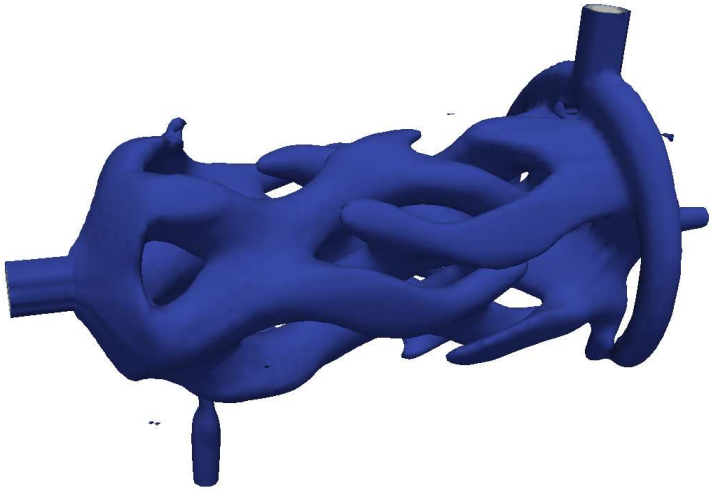


Figure 6.19. Velocity profile through the full-scale TPMS heat exchanger structure.

Another functionality of the program was a means of leak detection for the STL files of the full-scale TPMS heat exchangers. The algorithm to do this was a modified version of that used during the voxelisation of the TPMS unit cells. Beginning at the known primary fluid inlet location, the algorithm was seeded by allocating the fluid cells as belonging to the associated primary channel set. Any neighbouring fluid cells, as determined by the lattice adjacency rules, were also allocated to that channel set. Non-fluid cells were ignored, and created barriers beyond which fluids could remain unallocated, if there was no fluid path to reach them. This process was carried out over the entire spatial domain, and repeated for a number of iterations, until no further cells were allocated over an entire iteration. At this point, the fluid cells belonging to the secondary channel set were checked to ensure they had not been wrongly allocated to those of the primary, as they would if a leak were present. The algorithm was then repeated, using the secondary fluid inlet as the seed. After again checking the allocation of the cells at the opposite inlet location, the structure could be determined to be leak-free. This relatively simple algorithm was capable of checking the voxelised STL for errors in seconds, for the resolution at which the simulations were taking place. Although this process could not identify the more common issues related to the calibration and operation of the 3D printers themselves, checking the integrity of the original file could potentially solve issues before they eventuated into wasted prints.

The effectiveness of this algorithm could be verified by carrying out fluid-only, non-thermal simulations. The major alterations made to the program included disabling the modelling of

thermal effects, removing the inlet for the secondary fluid and an alteration to the initial conditions such that there were no fluid movements within the secondary fluid channels. Through these changes, any movement of fluid within the secondary channels, resulting in an observable velocity, would be an indication that a leak from the primary channels was present. If no fluid velocity within the secondary channels was recorded after the hydrodynamic lattice had attained a steady state, the design could be assumed to be leak free. This version of the program was greatly reduced in size and functionality, with the purpose of increasing computational speed. This additional simulation was largely unnecessary, since the same rules for fluid movements used by the hydrodynamic lattice were shared by the simpler algorithm.

The leak detection process is necessary largely due to the interfacial area between the packing and the heat exchanger entrances. With certain combinations of parameters for the shell, holes can develop between the packing edge and the shell, potentially allowing fluids to pass through. Commonly these holes are small, forming winding channels of only a single voxel in width.

This leak detection program is an implicit *a posteriori* method for determining the viability of a part after it has already been digitally created. Although the method does not directly locate holes within the STL mesh, their presence may be inferred from the results obtained. The holes may subsequently be removed by altering the parameters governing wall thickness and edge location. This may be contrasted with the use of programs such as Blender or MeshLab, which may be utilised in an explicit *a priori* manner, while the file is still being edited. These programs directly find the locations that holes may exist, and in some cases are able to repair them.

6.6 Discussion and Conclusions

Within the low Reynolds number, laminar flow regime tested, the Schoen gyroid structure has been found to provide the greatest heat transfer of the TPMS materials simulated. Rotations of the packing were found to make small differences to the heat transfer coefficients found, though not as large as the differences between the TPMS designs. The original orientation for each shape was mildly superior, and was determined to be a fair comparison point between the other geometries.

In theory, designs where a direct path is available, when the natural pores of the TPMS align with the direction of the primary axial flow, should have reduced heat transfer characteristics,

with the benefit of lowered pressure drop. Due to the low Péclet numbers in use however, not enough heat is transferred by advection that this channelling effect to become significant.

The Schwarz primitive TPMS design, while lower in heat transfer than the gyroid, provides much lower pressure drop, due to a lesser tortuosity of the structure. As seen in Figure 6.18, the heat transfer to pressure drop ratio was amongst the highest for the primitive, making the design viable in pressure limited applications. Although the Schwarz diamond design performed slightly worse than the gyroid in heat transfer, and worse than the primitive in pressure drop. The flat plate design has a lower surface area available to transfer heat between the fluid channels, though in the diffusive region investigated, this was not a significant issue. The flat plate outperformed most of the other designs in terms of Nusselt number, and was superior to all others when examining the Nusselt to Bejan ratio. One of the reasons for this is that the pressure drop appears, at such low Reynolds numbers, to be more influenced by geometry than the Nusselt number is. One possible explanation for this is that heat may still transfer through the walls regardless of the geometry. In the diffusive regime, the heat transfer will only be minorly disrupted. Conversely, fluid may not pass through the walls, so the pressure drop is a direct result of any and all obstructions to the flow.

The representative elemental volumes studied may be likened to the units making up a structured porous media. Across a number of studies of porous media, the Nusselt number has been compared across a range of Reynolds numbers. Although the data was placed between many of the correlations, there was not a good fit for any particular equation. This may be due to the TPMS designs requiring their own correlations, as different structures.

Although the LBM may be employed for high Reynolds flows in some cases (Yang *et al.*, 2012) with the appropriate treatment of turbulence, these situations often make use of relatively large length scales. When modelling advection-diffusion, the greatest stability may be found when the diffusion coefficient of the advection-diffusion lattice is much lower than the equivalent kinematic viscosity of the fluid. In such scenarios (Houlton, 2019), much lower resolutions, and greater Reynolds numbers may be used without issue, as a result of their much lower Schmidt number.

The thermal Péclet number is the ratio of the advective sources of heat transfer to the diffusive. The Péclet number used in this work is approximately $Pe = 2.4$, depending on the parameters being tested. At these low Péclet numbers, diffusive heat transfer is significant, and advective thermal transport is low. The influence of the specific TPMS shape effects is lessened, as the

advective terms are small. A majority of the heat is instead able to diffuse through the walls without being significantly impacted by the velocity profile. At higher Péclet numbers, it may be expected that shape factors will have a far greater effect, to the benefit of heat transfer ability. Therefore, more tortuous designs, such as the gyroid, which naturally have larger shape factors, will benefit more from higher Péclet number.

7 Conclusions

The objectives of this work were to produce a program capable of creating STL files suitable both for 3D printing and simulation; create a CFD code able to model unsteady three dimensional fluid flows; and the application of that code to test the heat transfer performance of a number of TPMS geometries capable of being included within a heat exchanger.

Heat exchanger geometries were generated in two parts in Chapter 3, through the use of a custom made MATLAB program. Unit cells of TPMS geometries were generated, and then formed into the channels of a heat exchanger structure. The goal was to measure the heat transfer ability of each geometry formed in this manner, making use of periodic boundaries to make the cell one typical shape, from many. The second part of this phase of the project was to generate full STL files which could then be used as the domain for further CFD, or sent to a 3D printer for physical creation of a full heat exchanger. These full heat exchanger designs incorporated both a TPMS internal structure, separated into two distinct flow channels, and a shell to contain the fluids. For the purposes of this project, a program was written to quickly and efficiently generate STL files of arbitrary mathematical functions, either in the form of unit cells, a part of a greater heat exchanger design, or a range of pre-defined shapes. Functions were also written to fill existing STL geometries with these same mathematical equations, creating new STL files in the process. This program reduced the part generation times from multiple days, down to seconds.

CFD simulations were carried out through the use of a 3D LBM conjugate heat transfer model. Two major versions were created. The first was a full-scale model, capable of determining the flow profile across the heat exchanger structures. This version allowed underperforming channels to be identified, and any leaks to be found. A second, representative elemental volume version was also created for more accurately modelling the heat transfer characteristics of the structures. This version utilised periodic boundaries to model an average unit cell from a greater structure. The computation speed of the LBM models was raised through the use of code parallelisation. Unsteady conjugate heat transfer is a developing field for the LBM, and was typically carried out previously with much simpler geometries, across two dimensions. Prior to the beginning of this work, no published work making use of LBM conjugate heat transfer across three dimensions could be identified in non-trivial conditions.

A selection of common TPMS morphologies were tested to determine those with the best potential to act as heat exchangers. This ability was quantified primarily through the Nusselt

number and dimensionless pressure drop in the form of the Bejan number for each of the designs. Numerical testing was carried out through the analysis of representative elemental volumes, made up of unit cells of the TPMS structures. The major results identified were that the influence that the inclusion of TPMS had on the shape factors affecting heat transfer were relatively minor at the low Reynolds and Péclet numbers studied. The addition of TPMS did impact the pressure drop experienced by the heat exchange structures, with the result that less tortuous designs were better. The pressure drop appears to be more heavily influenced by the geometry than the Nusselt number is, such that designs with less heat transfer ability, but much lower pressure drop were deemed generally better.

The highest rates of heat transfer, as measured by the largest Nusselt numbers, were found from the Schoen gyroid. Increasing the Reynolds number of the flow passing through the heat exchange structures had the effect of also increasing the Nusselt number in a linear manner. This trend can be expected to continue while the flow is in the laminar regime. TPMS geometries are currently being investigated across a wide range of chemical engineering fields. Heat transfer applications are particularly unexplored, the first results for which were published only recently.

The Schwarz primitive was found to be inferior to the gyroid for heat transfer ability under all the conditions tested. However, the primitive also had amongst the lowest of tortuosities and thus pressure drop. On a heat transfer per unit pressure drop basis, the primitive was the best of the TPMS designs trialled. It was surpassed in this metric as well by a flat plate construct, designed to minimise tortuosity and pressure drop.

The similarity in the performance of the TPMS designs is likely a result of the strongly diffusive fluid and heat movements at the conditions tested. In more turbulent flow regimes, it is likely that the shape factors caused by the TPMS geometries would have greater effects, and therefore a larger influence on the heat transferred.

7.1 Future work

The generation of STL files for use in the creation of the 3D printed TPMS heat exchangers is an area with large potential for improvements to be made. The STL files created for parts with small channels are large and can be very computationally expensive to produce. The most common method to reduce the file size of a given structure is to lower the number of triangles within the file, but doing so reduces the realism of the final print, and potentially introduces

non-manifold edges. Algorithmic improvements could provide alternatives to removing detail from the part. As discussed in Chapter 5, after an STL file is passed to the slicer software used by a 3D printer, the instructions necessary for the printer to construct the structure are created in the form of G-code. Within each unit cell of the structure, the G-code required is similar. Hence, for periodic regions with known dimensions, the STL file of only a single unit cell may be necessary, with the remaining parts simply extrapolated. In the case of TPMS and other geometries able to be expressed in simple mathematical forms, an STL file of the unit cell may not be needed at all. Another possibility is, with the appropriate *a priori* knowledge of the structures, the formulation of G-code directly from the TPMS equations. By these methods, the current file size limitations of 3D printers could be bypassed in the future, though this would not impact the computational difficulties of simulations for these structures.

The heat exchangers studied in the present work are based on the design of those parts actually able to be created through 3D printing in the past. As such, the numerical methods chosen were more appropriate for slow moving fluids passing through fine structures. Improvements to the 3D printing process are allowing for larger and more detailed parts to be created presently, while in the future, parts may become large enough for industrial applications. For flows of higher Reynolds number, the methods in this study may require adapting or replacing with either commercial packages such as Fluent, or alternative widely used open-source packages such as OpenFOAM.

Chemical reactions may be incorporated into the numerical method through the additions of a lattice for each solute species, and source terms governing accumulation and consumption due to the reaction. This process is relatively simple for species existing within the same fluid phase, but may be rapidly complicated by applications requiring multiple solvents. Due to the requirement of one lattice per species, the computational expense of such simulations may rise quickly. In addition, numerical stability becomes a greater challenge, particularly for species with greatly varying values of diffusivity coefficient. In a similar manner, a fouling process could be added to the simulations, whereby solute species are able to adsorb onto surfaces of the heat exchanger, and upon reaching a critical density, revoxelise the affected cell into a crystallised solid. Doing so would change the flow profile through the structure over time, as well as allowing for the inclusion of fouling terms within the calculation of the heat transfer coefficient.

Although the effects of creating new TPMS designs by carrying out arithmetic operations on combinations of the existing TPMS equations are discussed in Chapter 2, one variation of this

idea is to instead do so in a piecewise manner. Through 3D printing, the incorporation of discrete TPMS layers within a composite design becomes trivial to achieve. The potentially challenging portion of the design process is in the creation of leak-free STL files using such designs. Great care would be required at the join location to ensure that no leaks are created. As discussed in Chapter 2, such designs are already used in chromatography applications, but the implementation becomes more challenging for heat exchangers, due to the need to maintain separation of the fluid channels used. Baffle placement on the TPMS boundaries would be one possible solution to ensure fluid segregation. Such a design could be useful to reduce the pressure drop for regions of the heat exchanger where fluid movement is poor, to promote advection into these zones. Alternatively, regions with the greatest fluid movement, such as down the centre of the heat exchanger, may have improved heat transfer rates at the cost of additional pressure drop.

Many other variations for the overall design of the heat exchanger are possible. Through the repositioning of the channel entrances, addition of baffles, or inclusion of a better flow distributor, the effectiveness of the full-scale TPMS heat exchangers could likely be greatly raised. Each of these features would greatly alter the movement of fluids within the structures, favouring some areas, and moving the fluids away from others. A very large number of parameters are available to be investigated to optimise these systems. Many of these parameters are also difficult to study using the present representative elementary volume method.

8 References

Afshar, M., Anaraki, A. Pourkamali, Montazerian, H., & Kadkhodapour, J. (2016). Additive manufacturing and mechanical characterization of graded porosity scaffolds designed based on triply periodic minimal surface architectures. *Journal of the Mechanical Behavior of Biomedical Materials, 62*, 481-494. doi: 10.1016/j.jmbbm.2016.05.027

Alexander, F. J., Chen, H., Chen, S., & Doolen, G. D. (1992). Lattice Boltzmann model for compressible fluids. *Physical Review. A, Atomic, Molecular, and Optical Physics, 46*, 1967-1970. doi: 10.1103/physreva.46.1967

Alexander, F. J., Chen, S., & Sterling, J. D. (1993). Lattice Boltzmann thermohydrodynamics. *Physical Review E, 47*, R2249-R2252. doi: 10.1103/physreve.47.r2249

Ariza, Cristian, Casado, Cintia, Wang, Ruo-Qian, Adams, Eric, & Marugán, Javier. (2018). Comparative evaluation of OpenFOAM® and ANSYS® Fluent for the modeling of annular reactors. *Chemical Engineering & Technology, 41*, 1473-1483. doi: 10.1002/ceat.201700455

Ataee, Arash, Li, Yuncang, Fraser, Darren, Song, Guangsheng, & Wen, Cuie. (2018). Anisotropic Ti-6Al-4V gyroid scaffolds manufactured by electron beam melting (EBM) for bone implant applications. *Materials & Design, 137*, 345-354. doi: 10.1016/j.matdes.2017.10.040

Avgeropoulos, A., Dair, B. J., Hadjichristidis, N., & Thomas, E. L. (1997). Tricontinuous double gyroid cubic phase in triblock copolymers of the ABA type. *Macromolecules, 30*, 5634-5642. doi: 10.1021/ma970266z

Bejan, Adrian. (2004). Designed porous media: maximal heat transfer density at decreasing length scales. *International Journal of Heat and Mass Transfer, 47*, 3073-3083. doi: 10.1016/j.ijheatmasstransfer.2004.02.025

Bhatnagar, P. L., Gross, E. P., & Krook, M. (1954). A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Physical Review, 94*, 511-525. doi: 10.1103/PhysRev.94.511

Bhatti, S. K., Krishna, Ch M., Vundru, Ch, Neelapu, M. L., & Niranjan Kumar, I. N. (2006). *Estimating number of shells and determining the log mean temperature difference correction factor of shell and tube heat exchangers*. Paper presented at the Advanced Computational Methods in Heat Transfer IX.

Bieri, N. R., Chung, J., Haferl, S. E., Poulikakos, D., & Grigoropoulos, C. P. (2003). Microstructuring by printing and laser curing of nanoparticle solutions. *Applied Physics Letters, 82*, 3529-3531. doi: 10.1063/1.1575502

Buchanan, C., & Gardner, L. (2019). Metal 3D printing in construction: A review of methods, research, applications, opportunities and challenges. *Engineering structures, 180*, 332-348. doi: 10.1016/j.engstruct.2018.11.045

Bury, Tomasz, Składzień, Jan, & Widziewicz, Katarzyna. (2010). Experimental and numerical analyses of finned cross flow heat exchangers efficiency under non-uniform gas inlet flow conditions. *Archives of Thermodynamics, 31*, 133-144. doi: 10.2478/v10173-010-0034-5

Celik, Hasan, Mobedi, Moghtada, Manca, Oronzio, & Ozkol, Unver. (2017). A pore scale analysis for determination of interfacial convective heat transfer coefficient for thin periodic porous media under mixed convection. *International Journal of Numerical Methods for Heat & Fluid Flow, 27*, 2775-2798. doi: 10.1108/HFF-01-2017-0036

Chen, S., & Doolen, G. D. (1998). Lattice Boltzmann method for fluid flows. *Annual Review of Fluid Mechanics, 30*, 329-364. doi: 10.1146/annurev.fluid.30.1.329

Cheng, C. F., Hsueh, H. Y., Lai, C. H., Pan, C. J., Hwang, B. J., Hu, C. C., & Ho, R. M. (2015). Nanoporous gyroid platinum with high catalytic activity from block copolymer templates via electroless plating. *NPG Asia Materials, 7*. doi: 10.1038/am.2015.13

Cho, W. R., Wu, J. H., Shim, B. S., Kuan, W. F., Mastroianni, S. E., Young, W. S., . . . Martin, D. C. (2015). Synthesis and characterization of bicontinuous cubic poly(3,4-ethylene dioxythiophene) gyroid (PEDOT GYR) gels. *Physical Chemistry Chemical Physics, 17*, 5115-5123. doi: 10.1039/c4cp04426f

Cooperstein, I., Layani, M., & Magdassi, S. (2015). 3D printing of porous structures by UV-curable O/W emulsion for fabrication of conductive objects. *Journal of Materials Chemistry C, 3*, 2040-2044. doi: 10.1039/c4tc02215g

Crews, K., Huntley, C., Cooley, D., Phillips, B., & Curry, M. (2016). Influence of Cellulose on the Mechanical and Thermal Stability of ABS Plastic Composites. *International Journal of Polymer Science, 2016*, 1-10. doi: 10.1155/2016/9043086

Crossland, Edward J. W., Kamperman, Marleen, Nedelcu, Mihaela, Ducati, Caterina, Wiesner, Ulrich, Smilgies, Detlef M., . . . Snaith, Henry J. (2009). A Bicontinuous Double Gyroid Hybrid Solar Cell. *Nano letters, 9*, 2807-2812. doi: 10.1021/nl803174p

D'Orazio, A., & Succi, S. (2003). Boundary conditions for thermal lattice Boltzmann simulations. In M. A. P. Sloot, D. Abramson, A. V. Bogdanov, J. J. Dongarra, A. Y. Zomaya & Y. E. Gorbachev (Eds.), *Computational Science - ICCS 2003, Pt I, Proceedings* (Vol. 2657, pp. 977-986). Berlin: Springer-Verlag Berlin.

Davis, Mark E., & Davis, Robert J. (2003). *Fundamentals of Chemical Reaction Engineering* (International ed.). Boston: McGraw-Hill.

De Schampheleire, Sven, De Jaeger, Peter, Huisseune, Henk, Ameel, Bernd, T'Joen, Christophe, De Kerpel, Kathleen, & De Paepe, Michel. (2013). Thermal hydraulic performance of 10 PPI aluminium foam as alternative for louvered fins in an HVAC heat exchanger. *Applied Thermal Engineering, 51*, 371-382. doi: 10.1016/j.applthermaleng.2012.09.027

Di Ilio, G., Dorschner, B., Bella, G., Succi, S., & Karlin, I. V. (2018). Simulation of turbulent flows with the entropic multirelaxation time lattice Boltzmann method on body-fitted meshes. *Journal of Fluid Mechanics, 849*, 35-56. doi: 10.1017/jfm.2018.413

Dogonchi, A. S., & Ganji, D. D. (2016). Convection–radiation heat transfer study of moving fin with temperature-dependent thermal conductivity, heat transfer coefficient and heat generation. *Applied Thermal Engineering, 103*, 705-712. doi: 10.1016/j.applthermaleng.2016.04.121

Dolamore, Fabian. (2017). *In silico analysis of flow and dispersion in ordered porous media.* University of Canterbury. (Dissertation/Thesis)

Dolamore, Fabian, Fee, Conan, & Dimartino, Simone. (2018). Modelling ordered packed beds of spheres: The importance of bed orientation and the influence of tortuosity on dispersion. *Journal of Chromatography A, 1532*, 150-160. doi: 10.1016/j.chroma.2017.12.004

Eckel, Z. C., Zhou, C. Y., Martin, J. H., Jacobsen, A. J., Carter, W. B., & Schaedler, T. A. (2016). 3D printing additive manufacturing of polymer-derived ceramics. *Science, 351*, 58-62. doi: 10.1126/science.aad2688

Elliott, Olivia, Gray, Stephanie, McClay, Michael, Nassief, Bakr, Nunnelley, Ann, Vogt, Eric, . . . Carrano, Andres L. (2017). Design and Manufacturing of High Surface Area 3D-Printed Media for Moving Bed Bioreactors for Wastewater Treatment. *Journal of Contemporary Water Research & Education, 160*, 144-156. doi: 10.1111/j.1936-704X.2017.03246.x

Erbay, L. Berrin, Uğurlubilek, Nihal, Altun, Özge, & Doğan, Bahadır. (2017). Numerical investigation of the air-side thermal hydraulic performance of a louvered-fin and flat-tube heat exchanger at low Reynolds numbers. *Heat Transfer Engineering, 38*, 627-640. doi: 10.1080/01457632.2016.1200382

Fee, C., Nawada, S., & Dimartino, S. (2014). 3D printed porous media columns with fine control of column packing morphology. *Journal of Chromatography A, 1333*, 18-24. doi: 10.1016/j.chroma.2014.01.043

Feng, Jiawei, Fu, Jianzhong, Lin, Zhiwei, Shang, Ce, & Niu, Xiaomiao. (2019). Layered infill area generation from triply periodic minimal surfaces for additive manufacturing. *Computer aided design, 107*, 50-63. doi: 10.1016/j.cad.2018.09.005

Feng, Y. T., Han, K., & Owen, D. R. J. (2007). Coupled lattice Boltzmann method and discrete element modelling of particle transport in turbulent fluid flows: Computational issues. *International Journal for Numerical Methods in Engineering, 72*, 1111-1134. doi: 10.1002/nme.2114

Frederick, R. L., & Moraga, S. G. (2007). Three-dimensional natural convection in finned cubical enclosures. *International Journal of Heat and Fluid Flow, 28*, 289-298. doi: 10.1016/j.ijheatfluidflow.2006.03.005

Frisch, U., Hasslacher, B., & Pomeau, Y. (1986). Lattice-gas automata for the Navier-Stokes equation *Physical Review Letters, 56*, 1505-1508. doi: 10.1103/PhysRevLett.56.1505

Gandy, Paul J. F., Bardhan, Sonny, Mackay, Alan L., & Klinowski, Jacek. (2001). Nodal surface approximations to the P, G, D and I-WP triply periodic minimal surfaces. *Chemical Physics Letters, 336*, 187-195. doi: 10.1016/S0009-2614(00)01418-4

Große-Brauckmann, Karsten, & Wohlgemuth, Meinhard. (1996). The gyroid is embedded and has constant mean curvature companions. *Calculus of Variations and Partial Differential Equations, 4*, 499-523. doi: 10.1007/BF01261761

Hardy, J., Pomeau, Y., & de Pazzis, O. (1973). Time evolution of a two-dimensional model system. I. Invariant states and time correlation functions. *Journal of Mathematical Physics, 14*, 1746-1759. doi: 10.1063/1.1666248

He, X., Chen, S., & Doolen, G. D. (1998). A novel thermal model for the lattice Boltzmann method in incompressible limit. *Journal of Computational Physics, 146*, 282-300. doi: 10.1006/jcph.1998.6057

Houlton, Ben. (2019). *Enhancing packed bed geometry using computational fluid dynamic simulations.* University of Canterbury. (Dissertation/Thesis)

Hu, Kainan, Meng, Jianping, Zhang, Hongwu, Gu, Xiao-Jun, Emerson, David R., & Zhang, Yonghao. (2017). A comparative study of boundary conditions for lattice Boltzmann simulations of high Reynolds number flows. *Computers & Fluids, 156*, 1-8. doi: 10.1016/j.compfluid.2017.06.008

Huang, Samuel H., Liu, Peng, Mokasdar, Abhiram, & Hou, Liang. (2012). Additive manufacturing and its societal impact: a literature review. *The International Journal of Advanced Manufacturing Technology, 67*, 1191-1203. doi: 10.1007/s00170-012-4558-5

Huisseune, Henk, De Schampheleire, Sven, Ameel, Bernd, & De Paepe, Michel. (2015). Comparison of metal foam heat exchangers to a finned heat exchanger for low Reynolds number applications. *International Journal of Heat and Mass Transfer, 89*, 1-9. doi: 10.1016/j.ijheatmasstransfer.2015.05.013

Hyde, S. T., Andersson, S., Ericsson, B., & Larsson, K. (1984). A cubic structure consisting of a lipid bilayer forming an infinite periodic minimum surface of the gyroid type in the glycerolmonooleate-water system. *Zeitschrift Fur Kristallographie, 168*, 213-219. doi: 10.1524/zkri.1984.168.1-4.213

Imani, G., Maerefat, M., & Hooman, K. (2012). Lattice Boltzmann simulation of conjugate heat transfer from multiple heated obstacles mounted in a walled parallel plate channel. *Numerical Heat Transfer Part a-Applications, 62*, 798-821. doi: 10.1080/10407782.2012.709442

Imani, Gholamreza. (2017). Three dimensional lattice Boltzmann simulation of steady and transient finned natural convection problems with evaluation of different forcing and conjugate heat transfer schemes. *Computers & Mathematics with Applications, 74*, 1362-1378. doi: 10.1016/j.camwa.2017.06.024

Jung, Y., Chu, K. T., & Torquato, S. (2007). A variational level set approach for surface area minimization of triply-periodic surfaces. *Journal of Computational Physics, 223*, 711-730. doi: 10.1016/j.jcp.2006.10.007

Jung, Y., & Torquato, S. (2005). Fluid permeabilities of triply periodic minimal surfaces. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics, 72*, 056319. doi: 10.1103/PhysRevE.72.056319

Kadkhodapour, J., Montazerian, H., Darabi, A. Ch, Anaraki, A. P., Ahmadi, S. M., Zadpoor, A. A., & Schmauder, S. (2015). Failure mechanisms of additively manufactured porous biomaterials: Effects of porosity and type of unit cell. *Journal of the mechanical behavior of biomedical materials, 50*, 180-191. doi: 10.1016/j.jmbbm.2015.06.012

Kamiuto, Kouichi, & Yee, San San. (2005). Heat transfer correlations for open-cellular porous materials. *International Communications in Heat and Mass Transfer, 32*, 947-953. doi: 10.1016/j.icheatmasstransfer.2004.10.027

Karcher, Hermann, & Polthier, Konrad. (1996). Construction of Triply Periodic Minimal Surfaces. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical, and Engineering Sciences, 354*, 2077-2104. doi: 10.1098/rsta.1996.0093

Kellens, Karel, Baumers, Martin, Gutowski, Timothy G., Flanagan, William, Lifset, Reid, & Duflou, Joost R. (2017). Environmental dimensions of additive manufacturing: mapping application domains and their environmental implications. *Journal of Industrial Ecology, 21*, S49-S68. doi: 10.1111/jiec.12629

Khan, T. S., Khan, M. S., Chyu, Ming- C., & Ayub, Z. H. (2010). Experimental investigation of single phase convective heat transfer coefficient in a corrugated plate heat exchanger for multiple plate configurations. *Applied Thermal Engineering, 30*, 1058-1065. doi: 10.1016/j.applthermaleng.2010.01.021

Kruth, J. P. (1991). Material Incress Manufacturing by Rapid Prototyping Techniques. *CIRP Annals - Manufacturing Technology, 40*, 603-614. doi: 10.1016/S0007-8506(07)61136-6

Kuppan, T. (2013). *Heat Exchanger Design Handbook* (Second ed.). Boca Raton: CRC Press.

Kuruneru, S. T. W., Saha, S. C., Sauret, E., & Gu, Y. T. (2019). Transient heat transfer and non-isothermal particle-laden gas flows through porous metal foams of differing structure. *Applied Thermal Engineering, 150*, 888-903. doi: 10.1016/j.applthermaleng.2019.01.041

Kuwahara, Fujio, Shirota, Mitsuhiro, & Nakayama, Akira. (2001). A numerical study of interfacial convective heat transfer coefficient in two-energy equation model for convection in porous media. *International Journal of Heat and Mass Transfer, 44*, 1153-1159. doi: 10.1016/s0017-9310(00)00166-6

Lallemand, Pierre, & Luo, Li-Shi. (2003). Theory of the lattice Boltzmann method: acoustic and thermal properties in two and three dimensions. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics, 68*, 036706.

Lee, H. S. (2010). *Thermal Design: Heat Sinks, Thermoelectrics, Heat Pipes, Compact Heat Exchangers, and Solar Cells*. Hoboken: Wiley.

Leigh, S. J., Bradley, R. J., Purssell, C. P., Billson, D. R., & Hutchins, D. A. (2012). A Simple, Low-Cost Conductive Composite Material for 3D Printing of Electronic Sensors. *Plos One, 7*. doi: 10.1371/journal.pone.0049365

Ligon, Samuel Clark, Liska, Robert, Stampfl, Jürgen, Gurr, Matthias, & Mülhaupt, Rolf. (2017). Polymers for 3D printing and customized additive manufacturing. *Chemical Reviews, 117*, 10212-10290. doi: 10.1021/acs.chemrev.7b00074

Lopes, A. J., Lee, I. H., MacDonald, E., Quintana, R., & Wicker, R. (2014). Laser curing of silver-based conductive inks for in situ 3D structural electronics fabrication in stereolithography. *Journal of Materials Processing Technology, 214*, 1935-1945. doi: 10.1016/j.jmatprotec.2014.04.009

Lorensen, William E., & Cline, Harvey E. (1987). Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics (New York, N.Y.), 21*, 163-169. doi: 10.1145/37402.37422

Lu, J. H., Lei, H. Y., & Dai, C. S. (2017). A simple difference method for lattice Boltzmann algorithm to simulate conjugate heat transfer. *International Journal of Heat and Mass Transfer, 114*, 268-276. doi: 10.1016/j.ijheatmasstransfer.2017.05.115

Luo, Ji-Wang, Chen, Li, Min, Ting, Shan, Feng, Kang, Qinjun, & Tao, WenQuan. (2020). Macroscopic transport properties of Gyroid structures based on pore-scale studies:

Permeability, diffusivity and thermal conductivity. *International Journal of Heat and Mass Transfer, 146*, 118837. doi: 10.1016/j.ijheatmasstransfer.2019.118837

Ma, W. C., Huang, W. S., Ku, C. S., & Ho, R. M. (2016). Nanoporous gyroid metal oxides with controlled thickness and composition by atomic layer deposition from block copolymer templates. *Journal of Materials Chemistry C, 4*, 840-849. doi: 10.1039/c5tc02731d

Manglik, R. M., Sundén, Bengt, & Wang, L. (2007). *Plate Heat Exchangers: Design, Applications and Performance* (Vol. 11;1.;). Southampton;Boston;: WIT Press.

Matsen, M. W., & Schick, M. (1994). Stable and unstable phases of a diblock copolymer melt. *Physical Review Letters, 72*, 2660-2663. doi: 10.1103/PhysRevLett.72.2660

Matyka, Maciej, & Koza, Zbigniew. (2012). How to calculate tortuosity easily? *AIP Conference Proceedings, 1453*, 17-22. doi: 10.1063/1.4711147

Mayboudi, Layla S. (2020). *Comsol Heat Transfer Models*. Dulles, Virginia: Mercury Learning and Information.

McNamara, Guy R., Garcia, Alejandro L., & Alder, Berni J. (1995). Stabilization of thermal lattice Boltzmann models. *Journal of Statistical Physics, 81*, 395-408. doi: 10.1007/BF02179986

McNamara, Guy R., & Zanetti, Gianluigi. (1988). Use of the Boltzmann equation to simulate lattice-gas automata. *Physical Review Letters, 61*, 2332-2335. doi: 10.1103/physrevlett.61.2332

Meeks, William H., & Pérez, Joaquín. (2011). The classical theory of minimal surfaces. *Bulletin (new series) of the American Mathematical Society, 48*, 325-407. doi: 10.1090/S0273-0979-2011-01334-9

Mici, J., Rothenberg, B., Brisson, E., Wicks, S., & Stubbs, D. M. (2015). Optomechanical performance of 3D-printed mirrors with embedded cooling channels and substructures. In A. E. Hatheway (Ed.), *Optomechanical Engineering 2015* (Vol. 9573). Bellingham: Spie-Int Soc Optical Engineering.

Mohamad, A. A. (2019). *Lattice Boltzmann Method: Fundamentals and Engineering Applications with Computer Codes* (2nd 2019.;2; ed.). London: Springer London.

Montessori, Andrea, Lauricella, Marco, La Rocca, Michele, Succi, Sauro, Stolovicki, Elad, Ziblat, Roy, & Weitz, David. (2018). Regularized lattice Boltzmann multicomponent models for low capillary and Reynolds microfluidics flows. *Computers & Fluids, 167*, 33-39. doi: 10.1016/j.compfluid.2018.02.029

Moukalled, F., Mangani, L., Darwish, M., & SpringerLink. (2016). *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab* (1st 2016. ed. Vol. 113). Cham: Springer International Publishing.

Mukherjee, T., Zuback, J. S., De, A., & DebRoy, T. (2016). Printability of alloys for additive manufacturing. *International Journal of Scientific Reports, 6*, 19717. doi: 10.1038/srep19717

Najafi, Hamidreza, & Najafi, Behzad. (2010). Multi-objective optimization of a plate and frame heat exchanger via genetic algorithm. *Heat and Mass Transfer, 46*, 639-647. doi: 10.1007/s00231-010-0612-8

Najafi, Hamidreza, Najafi, Behzad, & Hoseinpoori, Pooya. (2011). Energy and cost optimization of a plate and fin heat exchanger using genetic algorithm. *Applied Thermal Engineering, 31*, 1839-1847. doi: 10.1016/j.applthermaleng.2011.02.031

Nakao, Andressa, Valdman, Andrea, Costa, André L. H., Bagajewicz, Miguel J., & Queiroz, Eduardo M. (2017). Incorporating Fouling Modeling into Shell-and-Tube Heat Exchanger Design. *Industrial & Engineering Chemistry Research, 56*, 4377-4385. doi: 10.1021/acs.iecr.6b03564

Nakayama, Akira, Ando, Kenji, Yang, Chen, Sano, Yoshihiko, Kuwahara, Fujio, & Liu, Jianjun. (2009). A study on interstitial heat transfer in consolidated and unconsolidated porous media. *Heat and Mass Transfer/Waerme- und Stoffuebertragung, 45*, 1365-1372. doi: 10.1007/s00231-009-0513-x

Osserman, Robert. (1986). *A Survey of Minimal Surfaces*. Toronto, Ontario: General Publishing Company.

Peng, Y., Shu, C., & Chew, Y. T. (2004). A 3D incompressible thermal lattice Boltzmann model and its application to simulate natural convection in a cubic cavity. *Journal of Computational Physics, 193*, 260-274. doi: 10.1016/j.jcp.2003.08.008

Perko, Janez, & Patel, Ravi A. (2014). Single-relaxation-time lattice Boltzmann scheme for advection-diffusion problems with large diffusion-coefficient heterogeneities and high-advection transport. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics, 89*, 053309. doi: 10.1103/PhysRevE.89.053309

Petrescu, Stoian. (1994). Comments on “The optimal spacing of parallel plates cooled by forced convection”. *International Journal of Heat and Mass Transfer, 37*, 1283-1283. doi: 10.1016/0017-9310(94)90213-5

Phillips, Carolyn L., Iacovella, Christopher R., & Glotzer, Sharon C. (2010). Stability of the double gyroid phase to nanoparticle polydispersity in polymer-tethered nanosphere systems. *Soft Matter, 6*, 1693-1703. doi: 10.1039/b911140a

Poddar, Tarun K., & Polley, Graham T. (2000). Optimize shell-and-tube heat exchanger design. *Chemical Engineering Progress, 96*, 41-46.

Saatdjian, E., Rodrigo, A. J. S., & Mota, J. P. B. (2011). Stokes flow heat transfer in an annular, rotating heat exchanger. *Applied Thermal Engineering, 31*, 1499-1507. doi: 10.1016/j.applthermaleng.2011.01.037

Saito, Shimpei, Abe, Yutaka, & Koyama, Kazuya. (2017). Lattice Boltzmann modeling and simulation of liquid jet breakup. *Physical Review. E, 96*, 013317-013317. doi: 10.1103/PhysRevE.96.013317

Scherer, Maik Rudolf Johann. (2013). *Double-Gyroid-Structured Functional Materials: Synthesis and Applications* (1 ed.). Heidelberg: Springer.

Schoen, A. H. (1970). Infinite periodic minimal surfaces without self-intersections. Washington D.C.: National Aeronautics and Space Administration.

Shah, R. K., & Sekulić, D. P. (1998). Nonuniform overall heat transfer coefficients in conventional heat exchanger design theory—Revisited. *Journal of Heat Transfer, 120*, 520-525. doi: 10.1115/1.2824281

Shamey, R., & Zhao, X. (2014). *Modelling, Simulation and Control of the Dyeing Process*: Elsevier Science.

Simon, Ursula, & Dimartino, Simone. (2019). Direct 3D printing of monolithic ion exchange adsorbers. *Journal of Chromatography A, 1587*, 119-128. doi: 10.1016/j.chroma.2018.12.017

Steinmann, Bettina. (2006). *Stereolithographic resins with high temperature and high impact resistance.* United States Patent No. 6989225. Washington D.C.: U.S. Patent and Trademark Office.

Sun, Jie, Peng, Zhuo, Zhou, Weibiao, Fuh, Jerry Y. H., Hong, Geok Soon, & Chiu, Annette. (2015). A review on 3D printing for customized food fabrication. *Procedia Manufacturing, 1*, 308-319. doi: 10.1016/j.promfg.2015.09.057

Symon, Nicholas. (2017). *Triply periodic minimal surface - based heat exchangers*. B.E. (Hons) Project Report. University of Canterbury.

Szost, B. A., Terzi, S., Martina, F., Boisselier, D., Prytuliak, A., Pining, T., . . . Jarvis, D. J. (2016). A comparative study of additive manufacturing techniques: Residual stress and microstructural analysis of CLAD and WAAM printed Ti-6Al-4V components. *Materials & Design, 89*, 559-567. doi: 10.1016/j.matdes.2015.09.115

Teruel, Federico E., & Díaz, Leonardo. (2013). Calculation of the interfacial heat transfer coefficient in porous media employing numerical simulations. *International Journal of Heat and Mass Transfer, 60*, 406-412. doi: 10.1016/j.ijheatmasstransfer.2012.12.022

Tu, Jiyuan, Yeoh, Guan Heng, Liu, Chaoqun, & ScienceDirect. (2012). *Computational fluid dynamics: a practical approach* (2nd ed.). Amsterdam: Butterworth-Heinemann.

Tubio, C. R., Azuaje, J., Escalante, L., Coelho, A., Guitian, F., Sotelo, E., & Gil, A. (2016). 3D printing of a heterogeneous copper-based catalyst. *Journal of Catalysis, 334*, 110-115.

Tubular Exchanger Manufacturers, Association. (1999). *Standards of the Tubular Exchanger Manufacturers Association* (8th ed.). Tarrytown, N.Y: Tubular Exchanger Manufacturers Association.

Urade, Vikrant N., Wei, Ta-Chen, Tate, Michael P., Kowalski, Jonathan D., & Hillhouse, Hugh W. (2007). Nanofabrication of double-gyroid thin films. *Chemistry of Materials, 19*, 768-777. doi: 10.1021/cm062136n

van der Hoef, Martin Anton, Beetstra, R., & Kuipers, J. A. M. (2005). Lattice-Boltzmann simulations of low-Reynolds-number flow past mono- and bidisperse arrays of spheres. *Journal of Fluid Mechanics, 528*, 233-254.

von Schnering, H. G., & Nesper, R. (1991). Nodal surfaces of Fourier series: fundamental invariants of structured matter. *Zeitschrift für Physik B Condensed Matter, 83*, 407-412. doi: 10.1007/BF01313411

Vukovic, Ivana, Voortman, Thomas P., Merino, Daniel Hermida, Portale, Giuseppe, Hiekkataipale, Panu, Ruokolainen, Janne, . . . Loos, Katja. (2012). Double gyroid network morphology in supramolecular diblock copolymer complexes. *Macromolecules, 45*, 3503-3512. doi: 10.1021/ma300273f

Wakao, Noriaki, & Kaguei, Seiichiro. (1982). *Heat and Mass Transfer in Packed Beds* (Vol. 1). New York: Gordon and Breach Science Publishers.

Wang, L., & Liu, J. (2014). Liquid phase 3D printing for quickly manufacturing conductive metal objects with low melting point alloy ink. *Science China-Technological Sciences, 57*, 1721-1728. doi: 10.1007/s11431-014-5583-4

Wang, Shaoxi, Yin, Yue, Hu, Chenxia, & Rezai, Pouya. (2018). 3D integrated circuit cooling with microfluidics. *Micromachines (Basel), 9*, 287. doi: 10.3390/mi9060287

Wang, Z. Y., Wang, J. J., Li, M. Y., Sun, K. H., & Liu, C. J. (2014). Three-dimensional printed acrylonitrile butadiene styrene framework coated with Cu-BTC metal-organic frameworks for the removal of methylene blue. *International Journal of Scientific Reports, 4*. doi: 10.1038/srep05939

Werner, J. G., Hoheisel, T. N., & Wiesner, U. (2014). Synthesis and characterization of gyroidal mesoporous carbons and carbon monoliths with tunable ultralarge pore size. *ACS Nano, 8*, 731-743. doi: 10.1021/nn405392t

Wu, H., Fahy, W. P., Kim, S., Kim, H., Zhao, N., Pilato, L., . . . Koo, J. H. (2020). Recent developments in polymers/polymer nanocomposites for additive manufacturing. *Progress in Materials Science, 111*, 100638. doi: 10.1016/j.pmatsci.2020.100638

Yang, Fan, Shi, Xuming, Guo, Xueyan, & Sai, Qingyi. (2012). MRT lattice Boltzmann schemes for high Reynolds number flow in two-dimensional lid-driven semi-circular cavity. *Energy Procedia, 16*, 639-644. doi: 10.1016/j.egypro.2012.01.103

Yang, Li, Hsu, Keng, Baughman, Brian, Godfrey, Donald, Medina, Francisco, Menon, Mamballykalathil, & Wiener, Soeren. (2017). *Additive Manufacturing of Metals: the Technology, Materials, Design and Production*. Cham: Springer International Publishing.

Yang, Zekun, Ma, Yingjie, Zhang, Nan, & Smith, Robin. (2020). Design optimization of shell and tube heat exchangers sizing with heat transfer enhancement. *Computers & Chemical Engineering, 137*, 106821. doi: 10.1016/j.compchemeng.2020.106821

Yoo, Dong-Jin. (2015). New paradigms in cellular material design and fabrication. *International Journal of Precision Engineering and Manufacturing, 16*, 2577-2589. doi: 10.1007/s12541-015-0330-8

Zhang, Junfeng. (2011). Lattice Boltzmann method for microfluidics: models and applications. *Microfluidics and Nanofluidics, 10*, 1-28. doi: 10.1007/s10404-010-0624-1

Zhang, Li-Zhi. (2013). *Conjugate Heat and Mass Transfer in Heat Mass Exchanger Ducts*: Academic Press.

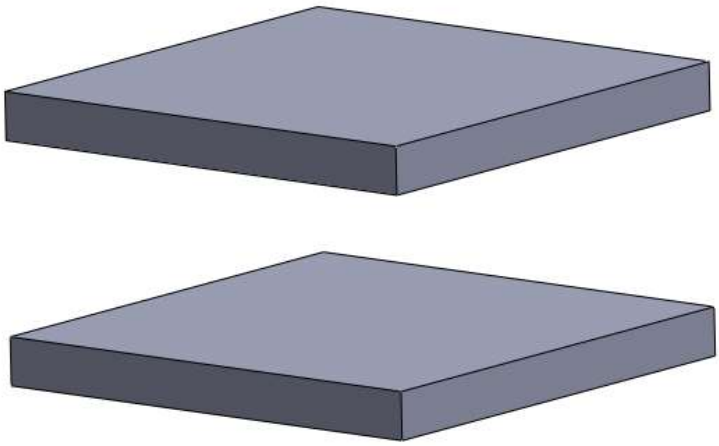
Zhao-Li, Guo, Chu-Guang, Zheng, & Bao-Chang, Shi. (2002). Non-equilibrium extrapolation method for velocity and pressure boundary conditions in the lattice Boltzmann method. *Chinese Physics, 11*, 366-374. doi: 10.1088/1009-1963/11/4/310

Zhao, Fuzhang. (2013). Optimal relaxation collisions for lattice Boltzmann methods. *Computers and Mathematics with Applications, 65*, 172-185. doi: 10.1016/j.camwa.2011.06.005

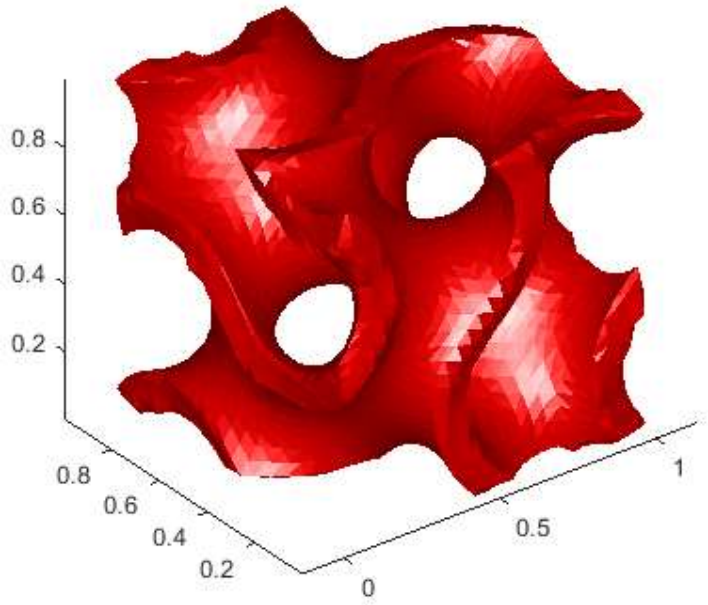
9 Appendices

Appendix A: Graphical examples of TPMS unit cells

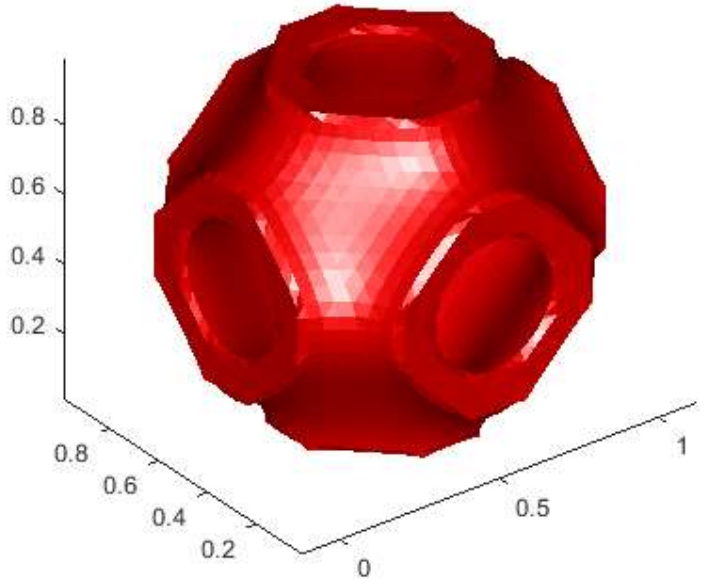
Flat Plate Example Unit Cell



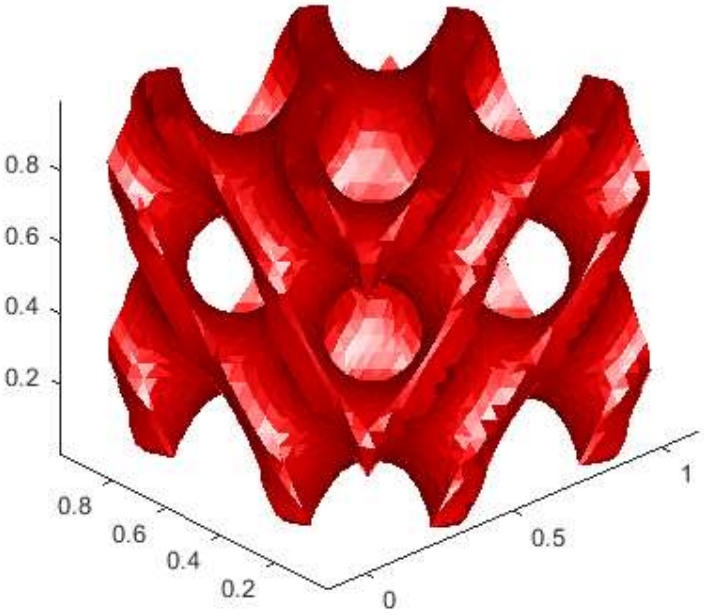
Gyroid Unit Cell



Primitive Unit Cell

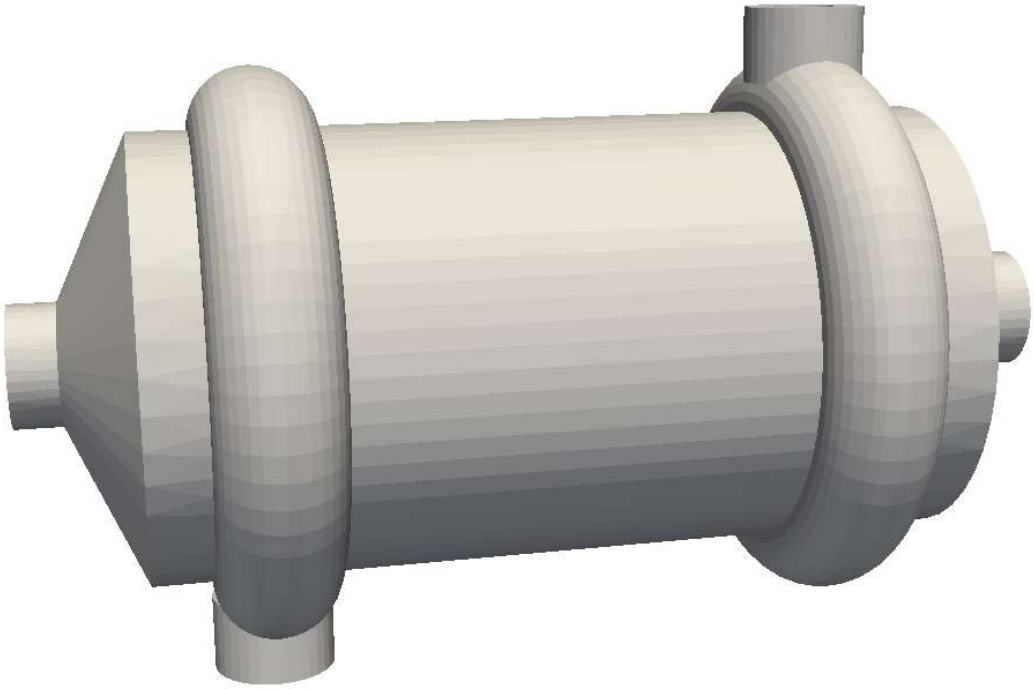


Diamond Unit Cell



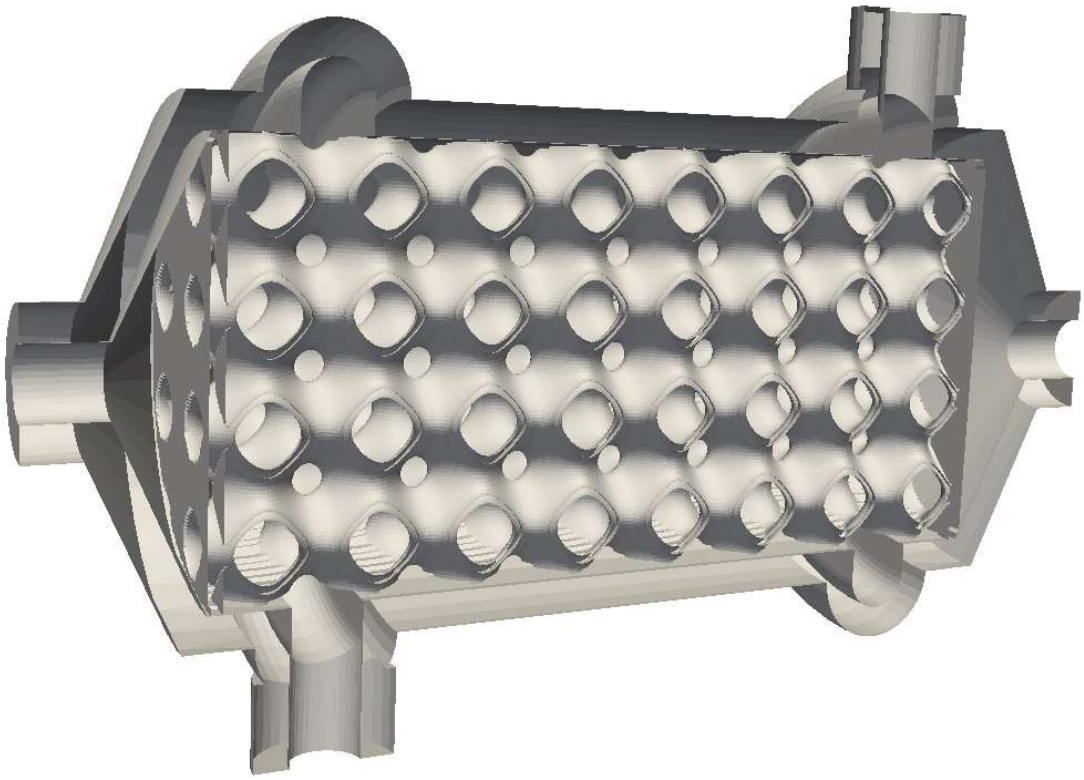
Appendix B: Full-scale TPMS heat exchanger designs

Full Heat Exchanger Design with Shell on

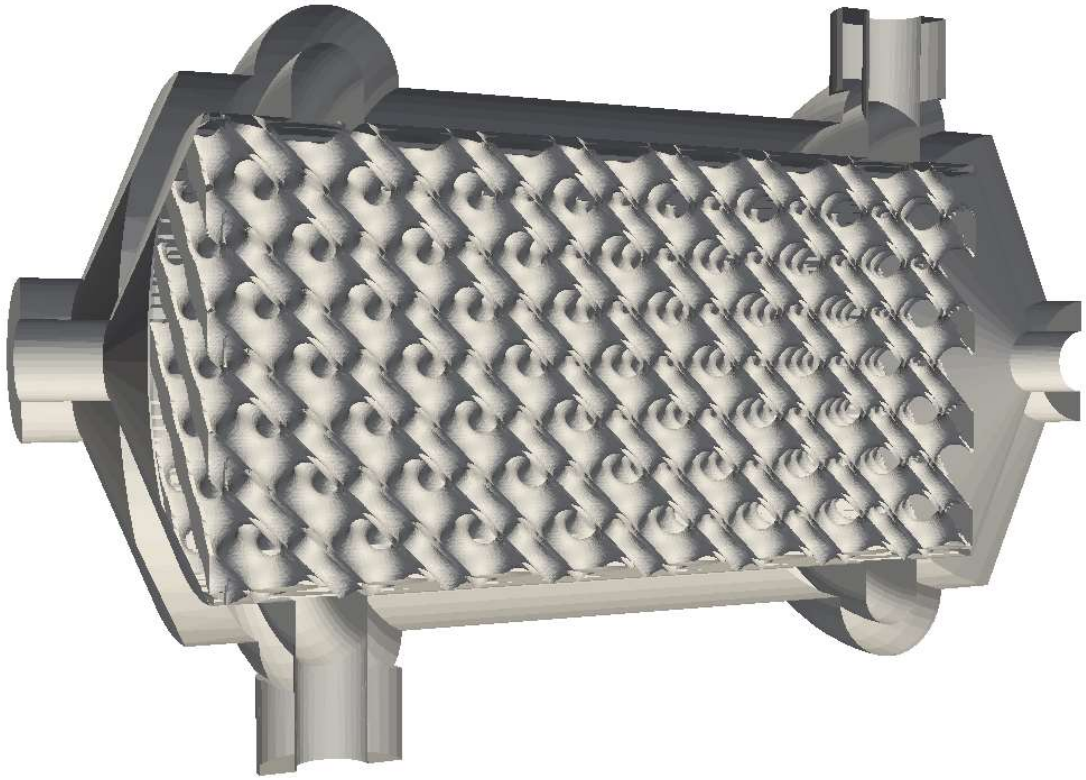


Gyroid Full Heat Exchanger

Primitive Full Heat Exchanger



Diamond Full Heat Exchanger

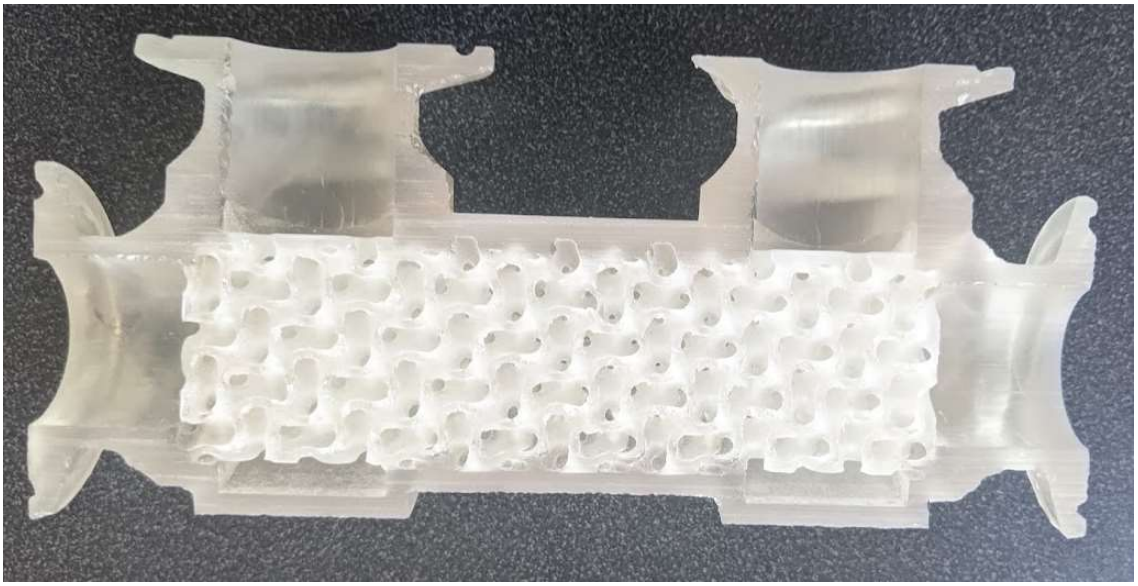


Appendix C: SLA printed parts

The following parts were printed on an LD-006 Creality resin SLA 3D printer. The macroscopic dimensions of the parts are 65x40x155 mm for the outer shell, while the internal structure is gyroidal, with wall thicknesses of 1 mm, and channel widths of 4 mm, adding up to full unit cells of 10 mm. The pictured heat exchangers have slightly different designs to those in Appendix B, having been through design changes to aid in printing.

Resin Gyroid Heat Exchanger

Resin Gyroid Heat Exchanger Cross Section with 45 Degree x Axis Rotation



Appendix D: Workflow diagram